AD-A251 557

[DOCU]MENTATION PAGE

Form Approved OMB No. 0704-0188

[...]s estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, [...]ing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this [...]ing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson [...]d to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

Field	Content
2. REPORT DATE	15 APR 92
3. REPORT TYPE AND DATES COVERED	Interim Report
4. TITLE AND SUBTITLE	Software and Design Document for the Rapid Acquisition of Manufactured Parts (RAMP) Small Mechanical Parts (SMP) Generative Process Planning Environment (GPPE) Version 1.5
5. FUNDING NUMBERS	C N000612-87-C-6000
6. AUTHOR(S)	N/A
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	South Carolina Research Authority 5300 International Boulevard N. Charleston, SC 29418
8. PERFORMING ORGANIZATION REPORT NUMBER	DDR003006-0
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	Naval Supply Systems Command Department of the Navy Washington, D.C.
10. SPONSORING/MONITORING AGENCY REPORT NUMBER	DDR003006-0
11. SUPPLEMENTARY NOTES	Prepared in cooperation with Naval Supply Sustems Command
12a. DISTRIBUTION/AVAILABILITY STATEMENT	~~DOD, DOE, NASA~~ This document has been approved for public release and sale; its distribution is unlimited.
12b. DISTRIBUTION CODE	

DTIC ELECTE JUN 19 1992 S A D

13. ABSTRACT (Maximum 200 words)

The Generative Process Planning Environment system (GPPE), is that component of the Rapid Acquisition of Manufactured Parts Small Mechanical Parts system that generates a Macro process plan for a part using generative methodologies. A typical macro process plan consists of a high-level workstation routing, and a Bill of Materials.

The purpose of this document is to describe the GPPE 1.5 design. This design description includes discussions of each major component of the GPPE. This component description includes a discussion of basic functionality, capabilities, and limitations.

Field	Content
14. SUBJECT TERMS	Bill of materials (BOM), Generative Process Planning Environment (GPPE), PDES, RAMP, SMP
15. NUMBER OF PAGES	60 front & back
16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT	UNCLASSIFIED
18. SECURITY CLASSIFICATION OF THIS PAGE	UNCLASSIFIED
19. SECURITY CLASSIFICATION OF ABSTRACT	UNCLASSIFIED
20. LIMITATION OF ABSTRACT	SAR

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

TABLE OF CONTENTS (CONTINUED)

Statement A per telecon Lorna Estep
Naval Supply Systems Command/SUP 642
Arlington, VA 22217-5000

NWW 6/18/92

TABLE OF CONTENTS (CONTINUED)

TABLE OF CONTENTS (CONTINUED)

This page intentionally left blank.

LIST OF FIGURES

This page intentionally left blank.

LIST OF TABLES

This page intentionally left blank.

LIST OF ACRONYMS AND ABBREVIATIONS

API	Applications Program Interface
BOM	Bill of Material
CAD	Computer Aided Design
CALS	Computer Aided Acquisition and Logistics Support
CASE	Computer Aided Software Engineering
CV	ComputerVision
DB	Database
ETPD	External Technical Product Data
GPPE	Generative Process Planning Environment
IAMS	Institute of Advanced Manufacturing Sciences
IDL	ICAD Design Language
IGES	Initial Graphics Exchange Specification
ITPD	Internal Technical Product Data
ME	Manufacturing Engineering
NC	Numerical Control
OIR	Organization for Industrial Research
P&IC	Production and Inventory Control
PDES	Product Data Exchange Using STEP
PIF	PDES Intermediate File
PST	Product Structure Tree
RAMP	Rapid Acquisition of Manufactured Parts
RPTS	RAMP Product Data Translation System
SCRA	South Carolina Research Authority
SMP	Small Mechanical Parts
STEP	Standard for the Exchange of Product Model Data
TDP	Technical Data Package

This page intentionally left blank.

RECORD OF CHANGE			
CHANGE NUMBER	CHANGE DATE	PAGES AFFECTED	COMMENTS

This page intentionally left blank.

SECTION 1.0
SCOPE

1.1 System Overview

The Generative Process Planning Environment system, hereafter referred to as the GPPE, is that component of the Rapid Acquisition of Manufactured Parts (RAMP) Small Mechanical Parts (SMP) system that generates a Macro process plan for a part using generative methodologies. A typical macro process plan consists of a high-level workstation routing, and a Bill of Material(s) (BOM).

1.2 Document Overview

The purpose of this document is to describe the GPPE 1.5 design. This design description includes discussions of each major component of the GPPE. This component description includes a discussion of basic functionality, capabilities and limitations.

GPPE is an object-oriented system (see 3.2.1 "GPPE Implementation Environment"). In an object-oriented environment, it is often difficult to characterize individual code unit inputs, outputs and processing. Indeed, it is often difficult to discriminate between the data and the manipulation of the data itself. In this document, the functionality, assumptions, and limitations, along with the expected inputs and outputs of each major component of the product structure tree (pst) will be discussed. (See Figure 3-1.)

Detailed descriptions of system usage and user interaction are not discussed in this document, but are discussed in detail in the Computer Operation Manual for the RAMP SMP Manufacturing System.

This document contains a number of figures depicting various components of the GPPE product structure tree. It also contains a number of appendices containing sample input and output data. In all cases, the figures and table data, along with any other specific examples, were derived from the same part instance (the actual part is depicted in Figure 3-5 "The Standard GPPE Screen Layout"). Thus, it is possible to follow the same part through the entire process planning sequence, from a raw Product Data Exchange Using Standard for the Exchange of Product Model Data (STEP) (PDES) file, to the completed process plan.

This page intentionally left blank.

SECTION 2.0
REFERENCED DOCUMENTS

The following documents (the exact issue shown) form a part of this specification to the extent specified herein.

2.1 RTIF Program Documents

Document No.	Date of Issue	Title
OMR003001-2	10 January 1992	Computer Operation Manual for the RAMP SMP Manufacturing System.
IMR001001-0 CH-2	13 February 1989	Generic Model for a RAMP Manufacturing System.

2.2 Other Documents

Title

The ICAD Browser User's Manual
ICAD, Inc.
201 Broadway, 7th Floor
Cambridge, Massachusetts 92139-1901

The ICAD System User's Manual (Vols 1 and 2)
ICAD, Inc.
201 Broadway, 7th Floor
Cambridge, Massachusetts 92139-1901

The ICAD Output Interface Toolkit
ICAD, Inc.
201 Broadway, 7th Floor
Cambridge, Massachusetts 92139-1901

IGES CAD Input Tools User's Manual
ICAD, Inc.
201 Broadwav, 7th Floor
Cambridge, Massachusetts 92139-1901

MetCAPP API Toolkit Programmers Reference Manual
Institute of Advanced Manufacturing Sciences, Inc.
1111 Edison Drive
Cincinnati, Ohio 45216

MetCAPP User's Guide
Institute of Advanced Manufacturing Sciences, Inc.
1111 Edison Drive
Cincinnati, Ohio 45216

This page intentionally left blank.

SECTION 3.0
DETAILED DESIGN

3.1 Purpose

The GPPE is the RAMP's SMP Generative Macro Process Planning function. As such GPPE is responsible for providing the following functions:

- o Import PDES part data from the PDES database.
- o Map from PDES form features to features suitable for manufacturing.
- o Determine process capabilities and available workstations applicable to the features of a part.
- o According to process sequencing rules, develop a flow through the shop floor workstations sufficient to manufacture a part.
- o Produce a workstation routing file, complete with estimated operation times computed from stored process time formulas and elemental time standards.
- o Produce a BOMs file, listing raw stock, tools, and other materials necessary for producing the final part.

(For more detail on specific functionality, refer to the "Generic Model for a RAMP Manufacturing System.")

3.2 Summary

The GPPE is an South Carolina Research Authority (SCRA)-developed application implemented within the ICAD environment. GPPE consists of six major components: the **PDES Design Model**, the **PDES Interpreter**, the **Manufacturing Part Model**, the **Resource Model**, the **Process model**, and the **Process Sequencer**. (Figure 3-1 illustrates these GPPE Product Structure Tree Components.) External databases provide resource and process information, while the MetCAPP system provides low level process planning knowledge.

GPPE takes as its main form of input, PDES data, and provides output in several forms. Its primary outputs are the process plan itself (the routing) and a BOM.

3.2.1 GPPE Implementation Environment

The GPPE is implemented within the ICAD environment. That is, the entire macro process planning application is written in ICAD's design language and Lisp. All functionality, from user-interface implementation to solids generation is accomplished within the ICAD environment (see Figure 3-2). The following sections explain the ICAD system, key ICAD concepts, and the role of ICAD in the GPPE implementation.

```
Generative Process Planning Environment
  Development-Pdes-Design-Model: Product-Item
    ...
  Pdes-Interpreter
    ...
  Mfg-Part-Model
    ...
  Resource-Model
    ...
  Process-Model
    ...
  Process-Sequencer
    ...
```

Figure 3-1 The GPPE Product Structure Tree Top-Level Components

3.2.1.1 The Product Model

ICAD is a knowledge-based engineering system. Techniques used to design, analyze and manufacture a product, are stored in a **product model**.

The ICAD product model represents the intent behind the product design. Within GPPE, the product model contains:

- product information: attributes of the physical product such as geometry and material type.
- resource information: the resources available for processing.
- process knowledge: knowledge of processes, such as metal removal operations and tool specifications, which are required to generate the process plan.
- process plan: attributes of the process plan including setup information, machines, sequenced operations, tools, lead-in and follow on operations.

In GPPE, the ICAD product model contains all the information required for the generation of the macro process plan.

Once all available product, resource, and process knowledge is collected and stored as an ICAD product model, a design may be evaluated by changing the input specifications (that is, provide new PDES data, resource data, or other user-specified inputs).

A product model is described in terms of its hierarchical product structure, that is, the overall application is broken down into components. This hierarchy is called the **product structure tree**. Each component of the product structure tree contains rules which characterize the component as well as the relationship between different components.

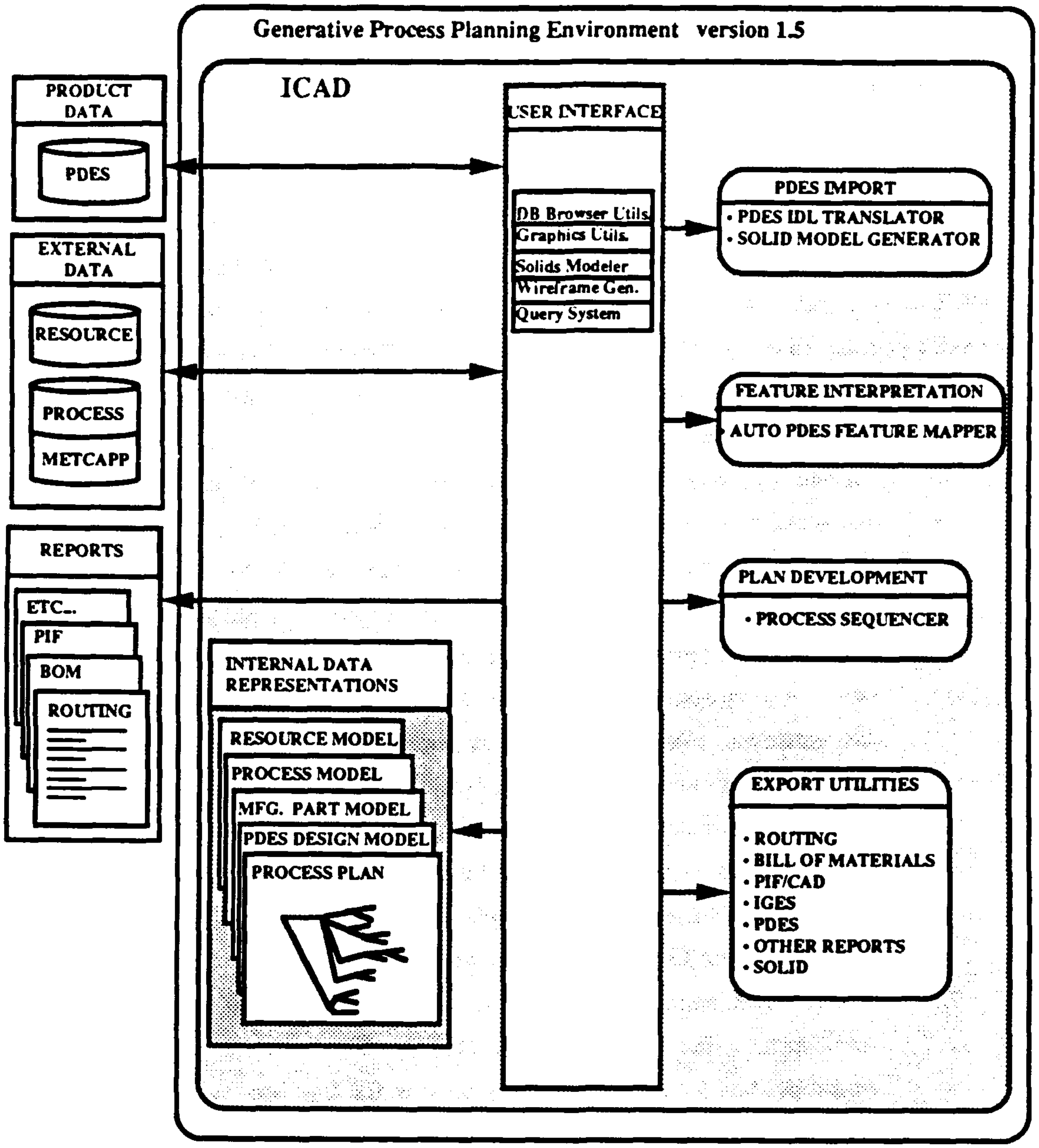


Figure 3-2 All GPPE Functions are Implemented within the ICAD Environment

3.2.1.2 The Development Language

All product model development is accomplished through the ICAD Design Language (IDL). IDL is an object-oriented programming language based on Common Lisp Flavors. Unlike a procedural programming language such as C or FORTRAN, in which procedures are defined for the manipulation of data, an object-oriented programming language defines classes of objects with their attributes. Therefore the basic form of an IDL program is an object which includes both data and instructions which manipulate that data.

3.2.1.2.1 The Defpart

The product model is effected through IDL code. The basic unit of description in IDL is called a **defpart**, or **definition** of a **part**. Defparts may be used to describe any aspect of a design or process. Defparts are used within GPPE to describe PDES models, the resource model, process plans, and every other aspect of the process planning environment. A typical defpart consists of five major components:

1) **Name**: the defpart name
2) **Mixins**: Mixins define "kind-of" relationships in which other defparts may become extensions of the current definition. Mixins reduce instances of duplicate code, and facilitate the rapid development of multiple part classes.
3) **Attributes**: Attributes are used to define the characteristics of an object. Attributes can describe an object's physical dimensions, its process requirements, its location in space, and any other characteristics. Some attributes are defined as constants, others are defined by expressions, or rules, that compute their values from other attribute values. Some attributes are defined as inputs, that is, their value comes from outside the defpart.
4) **Methods**: Methods are special attributes which are able to take arguments. Each time a method's value is requested, the method is evaluated and computes a value based on the argument which was supplied.
5) **Parts**: Parts create the product tree structure. Any defpart may contain zero or more parts. Each part defines a child or children to the parent. Each part minimally has a name, which will appear in the tree window, and a type, which identifies a defpart which contains the part's definition.

3.2.1.2.2 Instantiation

The ICAD system makes an important distinction between classes, which are generic descriptions of objects (for example, process plans), and instances, which are specific objects (for example, a process plan for a particular part). A defpart describes a class of objects, which is the set of possible instances that can be crea . from a single definition.

When an instance of a particular defpart is created, it is said that the defpart is **instantiated**. When a defpart is instantiated, all required attributes are computed. For example, instantiation of the PDES

Design Model component of GPPE involves extracting all data from the PDES data source specified as an input, and populating the PDES Design Model product structure tree with this data. The result is a fully populated model which represents a particular part, as defined by the incoming PDES data.

3.2.1.2.3 The Demand-driven Nature of IDL

When a defpart instance is created, the attributes' values are calculated according to a demand-driven evaluation mechanism. This means that attribute values are not calculated until they are required. The first time an attribute is required, its value is computed. Subsequent times when the attribute value is requested, the system returns the cached value and does not recompute it. IDL implements a robust consistency maintenance mechanism which assures that if any attribute is modified, all attributes which are either directly or indirectly affected by the change will be re-computed when demanded.

3.2.1.2.4 Protocols

In the GPPE environment, the concept of **protocols** is implemented. A protocol may be defined as the set of requirements which an object must fulfill to meet the demands of a given target application or usage. A protocol may define entity constraints and relationships, along with required and optional attributes. On a more concrete level, a protocol implemented in IDL code is a declaration of an object's essential characteristics.

Some protocols are very specific and apply only to a given branch of the product structure tree; for example, in the Manufacturing Part Model, each manufacturing feature instance is expected to conform to a solid model protocol, a wireframe protocol, and an output protocol (see 3.3.4.2 "Manufacturing Part Model Processing"). To conform to the solid model requirements, the definition of each feature must meet all attribute and other requirements required by the solid modeler to generate a solid. This includes the calculation of all required geometric feature parameters, the definition of one or more solid primitive parts, and a "solid" attribute which contains a list of all solid elements which make up the final feature solid.

Other protocol requirements are global, and are required of every object in the product structure tree. Most often these requirements apply to display characteristics and other user-interface demands. For example: every element in the GPPE product structure tree is expected to define a "specific-description- item" attribute. This attribute contains a list of critical values for the given object. This attribute is demanded by the "information" facility. When the "info" button is selected by the user, an information window containing data regarding the selected object is displayed on the screen.

By strictly adhering to specific protocols, code modifications and additions are greatly simplified. By understanding the protocol requirements of any given object type, it is a fairly straightforward process to define new objects which are immediately able to provide full functionality (see 3.3.2.2.1.2 "The PDES File Interface" for an example of this concept).

3.2.1.3 The ICAD User Interface Environment

ICAD provides a user-interface environment called the "ICAD Browser" in which all user interaction takes place. All user-interface capabilities within GPPE are implemented through the Browser. Within the Browser, the user may operate on the various GPPE elements to:

- o Specify input values.
- o View the product structure tree.
- o Create and manipulate the 3D solid or wireframe representation of a part.
- o View and modify the attribute values.
- o Create and review reports.
- o Debug part definitions.

3.2.1.4 External Data Interface

The ICAD environment provides the mechanism used by GPPE for representing and accessing tabular external data. This is described in the following sections.

3.2.1.4.1 Catalogs

The standard mechanism used for representing external data in the ICAD environment is the **catalog**. A catalog is a flat text file which may be used to characterize anything which can be described in tabular form (see Appendix IV for sample catalogs). Catalogs are interpreted by an ICAD module called the **peruser**. Perusing a catalog file makes the catalog available to the Relational Object Manager for query operations (see 3.2.1.4.2 "The Relational Object Manager"). Catalogs are generally perused at each system start-up. Currently, a mechanism does not exist which is capable of automatically recognizing that a catalog has been modified. Catalogs may however, be updated and manually perused at any time.

3.2.1.4.2 The Relational Object Manager

All data which is represented in catalogs is accessed through ICAD's Relational Object Manager (the ICAD query system). The ICAD query system permits catalog data to be accessed using standard relational algebra queries. Using these queries, catalog data may be selected, sorted, joined, or otherwise accessed and manipulated.

3.2.2 System Architectures

3.2.2.1 GPPE in the RAMP Architecture

The RAMP SMP Manufacturing Engineering (ME) architecture makes a formal distinction between "Macro Process Planning" and "Micro Process Planning" (see Figure 3-3). The objective of Macro process planning is to quickly plan the part in order to facilitate its scheduling and determine a projected delivery date to the customer. Thus, the purpose of GPPE is to produce a sequenced part routing with operation time standards and a BOM without developing a detailed process plan. In this way, enough plan detail is developed to allow Production and Inventory Control (P&IC) to assess a part's cost, schedule, and material requirements prior to the development of the detailed micro process plan. Similarly, GPPE's output is utilized by the Micro process planning system as a starting point for the generation of detailed

manufacturing instructions. GPPE accepts only PDES data as input. All other forms of input including Initial Graphic Exchange Specification (IGES) and paper drawings are currently processed through the exception process planning system.

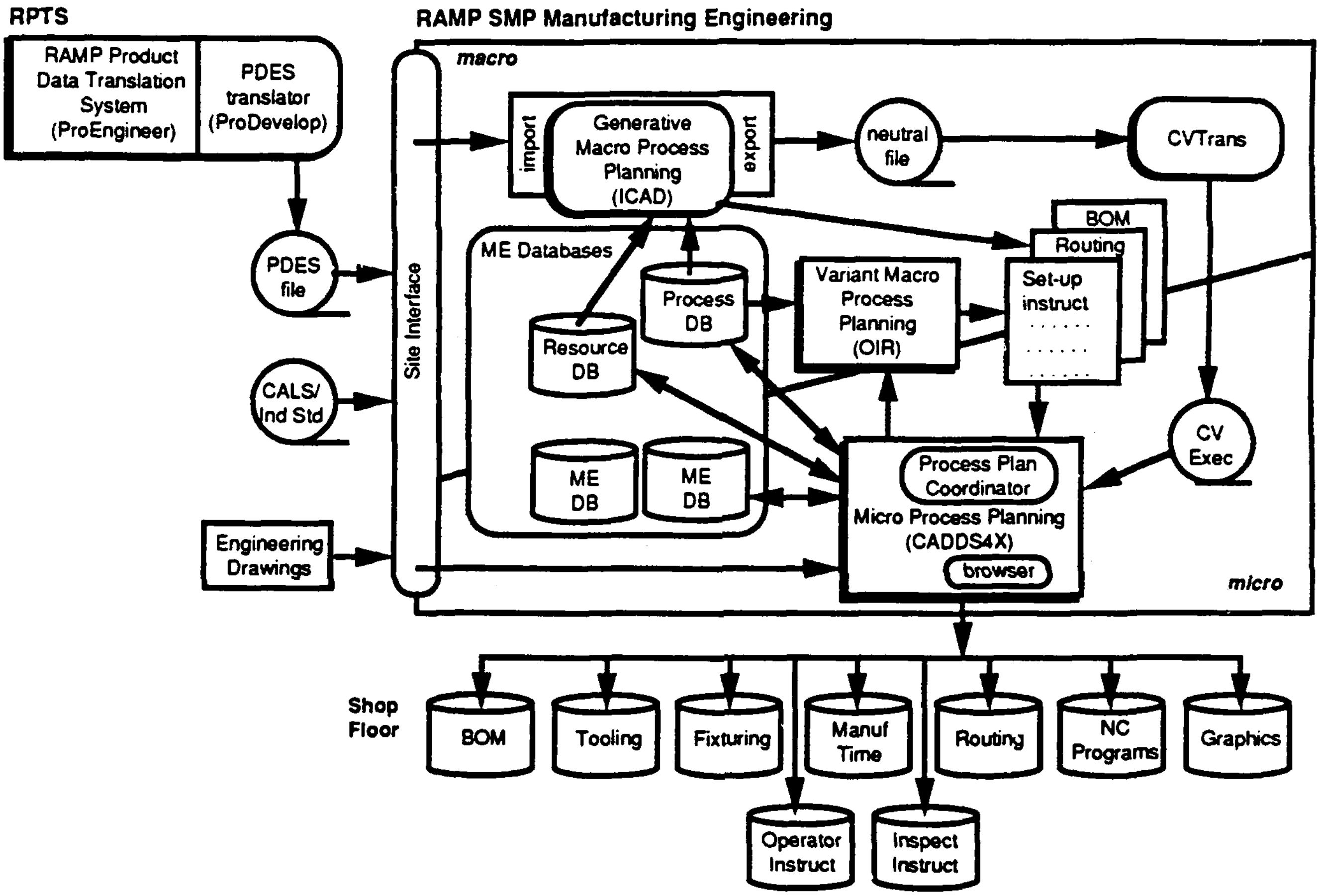


Figure 3-3 The RAMP SMP ME Architecture

3.2.2.2 The GPPE Architecture

The GPPE architecture consists of six major components (see Figure 3-4). These are:

- o The **PDES Design Model**, which is responsible for extracting information from the PDES data source and representing it in an internal format.
- o The **PDES Interpreter**, which is responsible for mapping from PDES form features to manufacturing features.

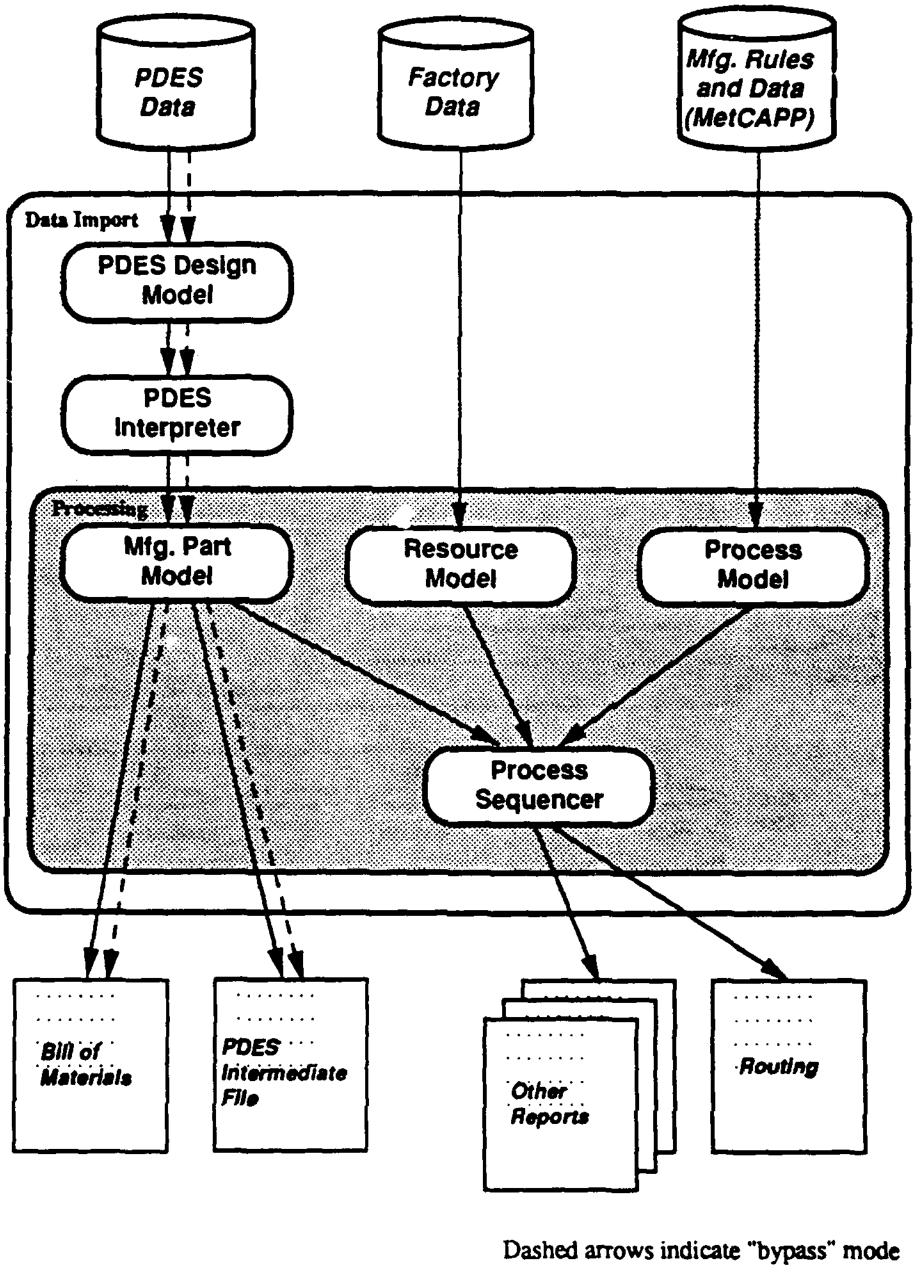


Figure 3-4 The GPPE Architecture

- The **Manufacturing Part Model**, which is primarily responsible for appropriately sizing and representing the manufacturing features. It is also responsible for generating wireframe and solid representations of the model, and for generating a PDES Intermediate File (PIF) which is further processed to generate an annotated solid on an external Computer Aided Design (CAD) system.
- The **Resource Model**, which is responsible for collecting and representing resource information such as in-stock materials, fixtures and workstations.
- The **Process Model**, which is responsible for representing process knowledge such as tool selection rules, operation derivation rules, material machinability data, process formulas, and elemental time standards.
- The **Process Sequencer**, which is responsible for providing a structured interactive environment in which all information from the other components may be considered and configured into a final process plan.

All elements and functions of GPPE, from the import of PDES data and the generation of graphics and process plans, to the user interface, are implemented within the ICAD environment.

3.3 Detailed Component Design

The following paragraphs describe the detailed design for each of the major components in the GPPE.

3.3.1 User Interface

3.3.1.1 User Interaction Within the ICAD Browser

The current GPPE user interface is built around the ICAD Browser, the standard ICAD production user environment (see 3.2.1.3 "The ICAD User Interface Environment"). Within the Browser window, the ME user manipulates a hierarchical data structure displayed graphically in a tree-like form (the "GPPE tree"). The main goal of the manufacturing engineer's activities is to develop a process plan sequence for the given part by directing the construction of a portion of the GPPE tree known as the **Process Sequencer**. The resulting sub-tree structure represents a hierarchically-specified process plan. From this structure a detailed textual representation of the process sequence (the "Macro Process Plan") is generated, which is then used by other downstream functions.

Action in the Browser takes place through a series of mouse clicks operating on the GPPE tree. There are two basic types of mouse actions--those that modify the tree structure (for example, expanding the children of an object) and those that demand output (such as a report). Both types of action invoke rules defined for the objects in the system. An example of a kind of rule which may be processed is one that prompts the user to enter information. In such a situation a pop-up window (called a "choice attribute window") appears on the screen, and the user types or selects the appropriate data which is then checked for its validity, and the pop-up is closed.

3.3.1.1.1 Building a Process Plan in the Browser Environment

There are several steps that the Manufacturing Engineer goes through to generate a process plan in the GPPE. Generally, the user begins a session to process a part order by expanding the GPPE sub-tree known as the Manufacturing Part Model, reviewing information about the specific manufacturing features, tolerances, notes and material specifications. Within the Manufacturing Part Model, he may use the solid modeling subsystem to generate a solid model of the part in order to visually verify the PDES data. The user verifies feature dimensions, tolerances, and notes by pressing the appropriate buttons at the bottom of the display. If the system indicates that the part is not supported by GPPE's own generative rules, then the user would generate a PIF, and would execute a "complete work" which signals the completion of the Macro process planning session.

If the part under consideration is supported by GPPE's generative rules, then the user would begin to build a process plan for the part. The main focus of the planning session is the expansion of the Process Sequencer sub-tree. It is here that the bulk of the interaction with the user takes place. A sequence of pop-up windows leads the user through the specification of tool and material preparation steps, workstation setups, selection of manufacturing processes for each part feature, ordering of the low-level machining operation requirements, selection and ordering of workstation time standards and follow-on operations until all part-processing requirements have been satisfied. At this point, the user selects the reports which are to be generated. These reports, which could include such items as the routing, a detailed routing description, the material requisition, and tool details, are generated and placed in the correct directories. At this point, the user is done, and may execute a "complete work."

3.3.1.2 The GPPE System Configuration

The GPPE configuration, (that is, the standard screen layout) consists of a graphics viewport, a view of the product structure tree, and a number of specialized buttons as shown in Figure 3-5. The standard configuration is stored and retrieved via the ICAD Browser configuration utility.

3.3.1.3 Customized Button Functions

The standard GPPE configuration displays many buttons on the screen which, when initiated, cause some action to take place. Special purpose buttons in the GPPE environment are defined through an ICAD function called "send-message-popping-result." This function allows a method to be defined in GPPE which performs any desired action. Common actions include:

- o Displaying an inspection window (for example, displaying a process plan for inspection): For each button of this type, there exists a method which collects all of the appropriate data via writers and companions (for a definition of writers and companions refer to 3.3.1.6 "Generating Text Output"), and displays it via the "message-popping-window" function. Windows displayed in this manner are

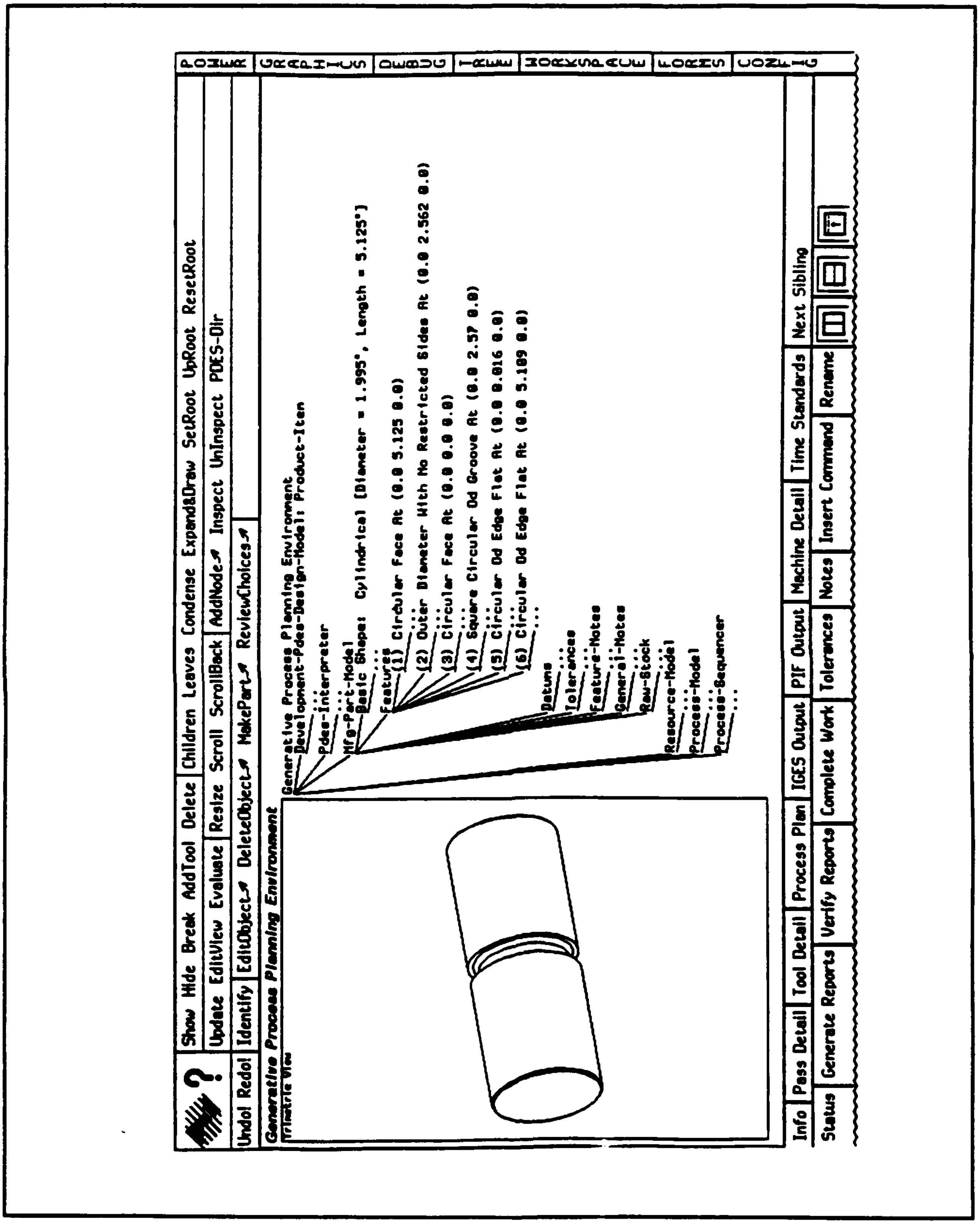


Figure 3-5 The Standard GPPE Screen Layout

scrollable, and may save their results in a user-specified file. The methods which perform these display functions are often mixed-in (see 3.2.1.2.1 "The Defpart") at different points in the product structure definition in order to provide inspection capabilities at various levels; for example, a PIF may be generated for an individual feature in the Manufacturing Part Model or from the entire GPPE product structure tree (see 3.3.4.2.3.2 "PIF Generation").

- Menu Selection (for example, generate reports menu): For each button of this type, there exists a method which makes a part which displays the appropriate options to the user via choice-attributes (see 3.3.1.4.1 "Text Input and Menu Selection"), and based on the user selection, performs the appropriate action.

3.3.1.4 Specifying Input Values

3.3.1.4.1 Text input and Menu Selection

The GPPE user is frequently prompted to select from a list of options or to enter data. This capability is implemented through the **choice-attributes** browser capability. Choice attributes allow free-form data entry in which the user may be required to enter numeric values, strings, or lisp constructs. Free form data is not automatically verified by choice attributes, but is verified by the calling function. Choice attributes are also utilized to present lists of items, from which zero, one, or many items may be selected by the user. Choice-attributes always provide an on-line help option.

3.3.1.4.2 Viewing the Product Structure Tree

The product structure tree is viewed and manipulated using built-in ICAD tree manipulation functions. The most commonly used functions include: "children" in which the children of the selected node are demanded and displayed, and "leaves" in which all of a selected node's descendants are demanded and displayed.

3.3.1.5 Solid and Wireframe Creation and Manipulation

ICAD utilizes the Parasolids modeling system for all solid generation and manipulation. ICAD uses its own built-in graphics routines to provide wireframe modeling capability. The solid or wireframe model may be displayed, rotated and manipulated via built-in ICAD utilities.

3.3.1.6 Viewing and Modifying Choices

Any inputs which are made by the user using the ICAD choice-attributes capability may be reviewed and modified. Modification of choice-attributes causes a re-evaluation of all affected attributes.

3.3.1.7 Generating Text Output

GPPE utilizes the ICAD Output Interface Toolkit for the generation of all reports. This toolkit allows the data embedded in an ICAD product model to be translated to virtually any desired format. Formatting is accomplished by using two critical constructs:

- Companions: A companion defines an additional view for a part from the perspective of the target system, in essence, extending the definition of the part;

For example, in GPPE a PIF companion exists for each manufacturing feature; each companion collects all necessary data to generate a PIF for the feature (see 3.3.4.2.3 "The PIF"). Companions may include special methods called "writer-methods" which are used to send messages to the writers.

- o Writers: A writer formats raw data sent to it by the product model (companions) and sends the formatted output to the target system interface, which could be a file, a window on the screen, or any other output device.

For more information regarding the Toolkit refer to "The ICAD Output Interface Toolkit" manual.

3.3.1.8 ICAD User Interface Limitations

The ICAD Browser is currently the sole source for all GPPE user interface capabilities. Accordingly, limitations in the ICAD Browser translate to limitations in the GPPE user interface. The following is a list of the more significant ICAD user interface limitations affecting GPPE system usability:

- o The graphical plan representation of the browser (that is, tree diagrams) is foreign to many ME users. Until production user interface capabilities are implemented, no other acceptable presentation format will be available through the ICAD Browser.
- o Limitations of the Browser restrict the user to presentation of the full part model via an unannotated graphical solid model and information from mouse gestures on the GPPE tree.
- o Because the Browser is inextricably tied to data structures which, by their nature, imply rigid, well-defined relationships where all data consistency and dependency relationships are maintained, it is not capable of offering a high degree of data manipulation flexibility. Due to this rigidity, it is often difficult, if not impossible, to perform several fundamental actions that are crucial in the construction of a structured plan (such as ordering, inserting, and deleting plan elements). Such capabilities require the implementation of a structured editing environment. This is difficult to accomplish in the current Browser interface environment. Although some of these capabilities can be implemented within the Browser environment, their implementation is not as straightforward as might be desired.

3.3.2 PDES Design Model Summary

GPPE's sole source of product information is in the form of PDES data (see Appendix I for a sample PDES file). This external PDES data (the External Technical Product Data (ETPD)) is incorporated into GPPE to create a hierarchical structure called the PDES Design Model, which contains an internal representation of the original PDES data (the Internal Technical Product Data (ITPD)). This structure is hierarchical in nature, and consists of an ICAD representation of the original PDES data (see Figure 3-6). The ITPD is the source of all product data information for all other GPPE components.

```
Development-Pdes-Design-Model: Product-Item
  Version 1: Product-Item-Version
    Creator: Person-And-Organization
      ...
    Creation-Date: Date-Time
      ...
    Definition 1: Product-Item-Version-Functional-Definition
      Specific-Functionality: Part-Model
        Model-Shape: Manifold-Solid-Brep
          ...
        Model-Units: Units
          ...
        Nominal-Shape: Shape
          ...
        Part-Feature 1: Form-Feature
          ...
        Part-Feature 2: Form-Feature
          ...
        Part-Feature 3: Form-Feature
          ...
        Part-Feature 4: Form-Feature
          ...
        Part-Dimension 1: Location-Dimension
          ...
        Part-Dimension 2: Size-Characteristic-Dimension
          ...
        Part-Dimension 3: Location-Dimension
          ...
        Part-Dimension 4: Location-Dimension
          ...
      Material 1: Material-Property
        ...
      Notes-And-Spec 1: General-Note = "Break All Sharp Edges"
        ...
      Notes-And-Spec 2: Chemical-Property
        ...
      Notes-And-Spec 3: Heat-Treat
        ...
      Notes-And-Spec 4: Inspection-Property
        ...
      Notes-And-Spec 5: General-Note = "Stock Is Nominal Size W
        ...
      Notes-And-Spec 6: General-Note = "Dimension From One End (
        ...
      Notes-And-Spec 7: General-Note = "Dimension From Opposite
        ...
      Owner: Person-And-Organization
        ...
      Creation-Date: Date-Time
        ...
```

Figure 3-6 ICAD Internal Representation of PDES data

3.3.2.1 PDES Design Model Inputs

3.3.2.1.1 Order Number

The order number is a ten digit user-supplied number. The order number is used to identify a **defspec** file of the same name (see Figure 3-7) which contains all of the necessary PDES source information. A defspec is a built-in ICAD construct which may be used to initialize a part. The part-number and revision-number of the GPPE defspec are used to access the appropriate PDES data.

```
(in-package 'RAMP)

(def-spec R0000000179 GPPE
 :order-number "0000000179"
 :national-stock-number "5432101234567"
 :part-number "80020_423498                "
 :revision-number "NR "
 :quantity 2
 )
```

Figure 3-7 Sample Defspec fo. ~rder Number "0000000179"

3.3.2.1.2 PDES Data

3.3.2.1.2.1 Implicit Feature Representation

The current version of GPPE uses a RAMP version of PDES that relies extensively on the use of **implicit form features**. A PDES implicit form feature represents a stereotypical portion of a shape. Some common shapes would include holes, pockets, walls, and grooves. Implicit form features model shape information parametrically rather than geometrically. For example, a hole might be described by giving a diameter and center axis rather than a cylindrical surface. The use of implicit form features simplifies the problem of specifying and communicating a product design.

3.3.2.1.2.2 PDES Data Sources

GPPE is capable of reading PDES data either from a flat text file off disk, or from an Oracle database.

3.3.2.1.2.2.1 ORACLE database representation

In the RAMP production environment, PDES data is stored in an ORACLE database.

3.3.2.1.2.2.1.1 Extracting PDES Data from the Oracle Database

PDES data is extracted from the database via the CREATE-ICAD-FILE routine. CREATE-ICAD-FILE performs two specific functions:

1) Extract PDES data: The a~propriate Oracle qu.ry operations are performed to extract the PDES data fro ne database.

2) Format the data: The resulting PDES data is formatted. This format is not the same as the original raw PDES file format. Parentheses are placed around each PDES element, and each field in the file is parsed and expanded with its associated type and value (see Appendix II for a sample file).

3.3.2.1.2.2.2 Raw PDES file

A raw PDES flat text file (see Appendix I for a sample raw PDES file) may be read and placed into the PDES Design Model structure. Because of RAMP restrictions, this capability is currently used only in the development environment.

3.3.2.2 PDES Design Model Instantiation

Instantiation of the PDES design model is accomplished through a PDES import facility. This facility is responsible for accessing the PDES data file, parsing it, and converting the external PDES data structure into a hierarchical internal format (see Figure 3-6).

3.3.2.2.1 The PDES Import Facility

GPPE implements an object-oriented PDES reader. This reader parses the PDES file, and creates the PDES Design Model structure based on the PDES feature protocols.

3.3.2.2.1.1 The PDES File Interface

The PDES file interface parses PDES data, line-by-line, and puts it into a hash table. This hash table is later used to instantiate the PDES Design Model with the appropriate values. The parser functions in the following manner:

```
Begin
      Create hash-table
      For each line in the PDES file
      begin
        If the PDES data was not originated by CREATE-ICAD-FILE
              begin
                 Parse data to identify a key (the line number), type, and the data
                 For each raw-data element
                   begin
                       clean up data that is, remove underscores exchange .T. for t.
                       populate hash table with the clean data
                    end
        else (data already parsed and cleaned)
                 For each data element in the parsed line
                   begin
                       populate hash table with data
                   end
        endif
      end
End
```

3.3.2.2.1.2 Creating the PDES Internal Representation

3.3.2.2.1.2.1 Automatic Code Generation Using Expresstool

In GPPE, the IDL representation of the PDES entity definitions are generated automatically from entity definitions modeled using the Express language. Express is the data modeling language in which the PDES specification is modeled, and as such is the central data definition language driving the import of PDES data to GPPE's internal representation.

The automatic generation of IDL code representing PDES entity definitions is accomplished by utilizing a customized version of Versant Object Technology Corp.'s **Expresstool**. ExpressTool is a Unix-based Computer Aided Software Engineering (CASE) utility that enables the automatic generation and configuration control of application interfaces based on data modeled in the Express language, most notably the PDES specification. A set of library functions allows quick user customization of application-specific formatting modules that generate, directly from Express, a wide range of data representations and interfaces.

In the GPPE system, Expresstool takes as input, a set of Express definitions which define the PDES model; these are processed, and an IDL representation is generated. This automatic code generation simplifies system maintenance, especially as wholesale PDES modeling changes are implemented.

As an example, the Express definition of an implicit-knurl form feature shown in Figure 3-8 would be translated into the IDL representation of the same feature depicted in Figure 3-9.

```
ENTITY IMPLICIT-KNURL
        SUPERTYPE OF (implicit_diagonal_knurl XOR
                      implicit_diamond_knuurl XOR
                      implicit_straight_knurl)
        SUBTYPE OF (implicit-area-feature);
        number_of_teeth          : INTEGER
        knurl_major_dim      : size_parameter;
        knurl_nominal_dim    : size_parameter;
        tooth_depth          : size_parameter;
        fillet_at_root       : size_parameter;
   WHERE
        cylindrical (installation_area);
END_ENTITY;
```

Figure 3-8 Express definition for an Implicit-Knurl Form Feature

```
(define-entity IMPLICIT-KNURL (implicit-area-feature)
 :attributes (:attribute-order '(:installation-area
                                   :number-of-teeth :knurl-major-dim :knurl-nominal-dim
                                   :tooth-depth :fillet-at-root)
          :number-of-teeth (the (:attribute-value :number-of-teeth)) ;INTEGER
      )
 :parts ((knurl-major-dim :type
              (the (:subentity-type :knurl-major-dim))        ;SIZE-PARAMETER
              :entity-id (the (:subentity-id :knurl-major-dim))
                          )
         (knurl-nominal-dim :type
              (the (:subentity-type :knurl-nominal-dim)) ;SIZE-PARAMETER
              :entity-id (the (:subentity-id :knurl-nominal-dim))
                          )
         (tooth-depth :type
              (the (:subentity-type :tooth-depth))     ;SIZE-PARAMETER
              :entity-id (the (:subentity-id :tooth-depth))
                      )
         (fillet-at-root :type
              (the (:subentity-type :fillet-at-root))    ;SIZE-PARAMETER
              :entity-id (the (:subentity-id :fillet-at-root))
                          )
     )
 )
```

Figure 3-9 IDL definition for an Implicit-Knurl Form Feature

3.3.2.2.1.3 Feature Protocol Processing

The IDL feature protocol (as shown in Figure 3-9) is used in the feature import process to define the structure of the PDES Design Model and to populate it with the correct values. The "define-entity" macro utilized in the IDL feature protocol expands into two other IDL constructs, an IDL defpart, and an interpreter mixin defined for that defpart. This enables user-defined extensions to the feature protocol to be easily added within the PDES Interpreter construct.

The PDES Design Model is instantiated in the following manner:

```
instantiate (object)
    begin
        if there are descendants
            begin
                compute attributes for self
                instantiate all descendants
            end
        else
            compute attributes for self
    end
```

Where the "object" first given to "instantiate" is the PDES "Product-Model" (the Product-Model sits at the top of the PDES structure; as such, all other entities and others are its descendants).

The existence of descendants, the number of descendants, and their attribute values is determined by the IDL feature definition protocol defined above and by the hash table which contains the PDES data.

3.3.2.3 PDES Design Model Limitations and Assumptions

3.3.2.3.1 State of Incoming Data

It is assumed that the incoming PDES data has been validated as being syntactically and semantically correct, and that the data is complete. Minimal in-line validation of PDES data is currently performed by GPPE.

Units are assumed to be in inches throughout the system. There is no checking of the PDES units entities upon input.

3.3.2.3.2 RAMP PDES Implementation

Successful PDES interpretation (see 3.3.3 "PDES Interpreter Summary") depends on RAMP's PDES implementation, the semantics of which are rigid in many ways, including:

- o implicit form features: an implicit form feature approach is used, as opposed to an explicit approach of a more low-level topological or geometric nature
- o shape model representation: the shape model must be a topological model of the basic shape only.

3.3.2.3.3 Infinite PDES Design Model Product Structure Tree

In the current implementation of the SMP PDES Generation System, PDES end-bound features are not represented correctly. Some form features are defined in PDES by a feature definition and an end-bound. End bounds refer to one or more other features which may have their own end-bounds. This can lead to a circular feature definition. If the user performs a "leaves" (expand the tree to its limit) operation on a feature defined in this manner, the system will enter an infinite recursion and will possibly use up all available memory. However, because the PDES Design Model is not displayed in the production user environment, this limitation is never apparent to the user.

3.3.2.4 PDES Design Model Outputs

The PDES Design Model has no real outputs in the standard sense, but exists simply as an internal representation of PDES data. The PDES Design Model responds to requests for attributes of the elements which it contains (see 3.2.1.2.3 "The Demand-driven nature of IDL"). Thus the outputs of the PDES Design Model may be considered to be all information about the current PDES part instance.

3.3.3 PDES Interpreter Summary

In the manufacturing domain, process planning involves the selection and sequencing of operations and processes necessary to transform raw material into a finished part. Before process planning knowledge can be characterized, the objects to which that knowledge refers must be identified and represented. These objects are known as **manufacturing features**. These features are useful for process

planning because they serve to classify geometric and topological patterns as being manufacturable by one process or another. A feature decomposition for one particular part in essence comprises a very high-level process plan for the manufacture of that part. Thus the act of classifying a part region as being of one type of manufacturing feature or another is itself a process planning action.

The PDES Interpreter serves the purpose of interpreting or mapping PDES data as represented in the PDES Design Model into a common format for which a process plan can be generated. In the case of PDES form features, the function of the PDES Interpreter is to perform a mapping from a form feature representation to a manufacturing feature representation suitable for process plan generation. In the case of all other PDES data (such as materials, notes and datums) the function of the PDES Interpreter is simply to extract all pertinent information from the PDES Design Model so that it may be used in the process planning operation.

3.3.3.1 PDES Interpreter Inputs

The PDES Interpreter uses the PDES Design Model as its sole data source. Individual attribute values are requested from the PDES Design Model on an as-needed basis. There are no other external inputs.

3.3.3.2 PDES Interpreter Instantiation

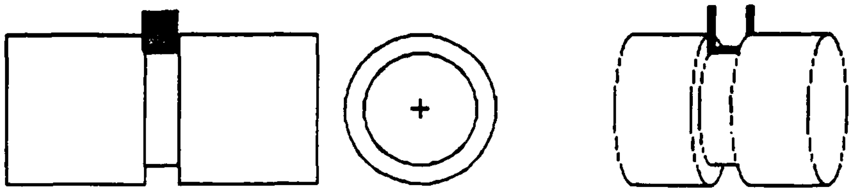
3.3.3.2.1 Feature Mapping

The GPPE converts the PDES implicit form features of the PDES Design Model instance into explicit manufacturing features. This one-to-many mapping is accomplished by taking each form feature's shape, its application to the product, and its relationship to other features into account, and applying rules which use this information to determine the manufacturing feature mapping. For example, a PDES "square-u-profile" applied on a "linear" path would be mapped into a "square-linear-groove" manufacturing feature (see Figure 3-10 for a sample feature mapping).

The same PDES implicit feature, when applied in differing orientations or with differing coincident geometry can result in a different manufacturing feature mapping. For example, an edge flat applied around an outer edge of a turned part implies one manufacturing feature mapping (a chamfer), while an edge flat applied at the top of a hole would imply a different mapping (a countersink). In order to provide a consistent, accurate feature mapping capability, each different feature orientation/application combination must be accounted for individually.

3.3.3.2.1.1 Methodology for Mapping of PDES Elements

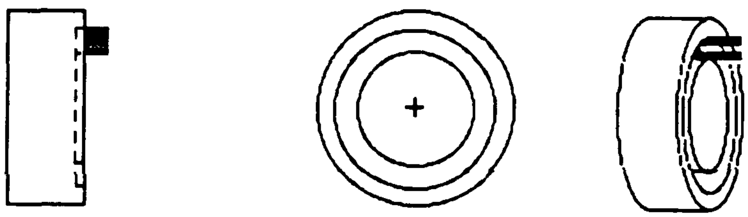
The PDES Interpreter utilizes a single methodology for mapping from PDES Design Model form features to manufacturing features. There are three important steps in this methodology:



PDES FORM FEATURE: DEPRESSION - CIRCULAR SQUARE U
*(PROFILE: SQUARE-U
PATH: CIRCULAR
PATH ANGLE: 90)*

WOULD BE MAPPED INTO ONE OF:

GPPE MANUFACTURING FEATURE:
1) SQUARE-U CIRCULAR OD GROOVE
2) OD WITH TWO RESTRICTED SIDES



PDES FORM FEATURE: DEPRESSION - CIRCULAR SQUARE U
*(PROFILE: SQUARE-U
PATH: CIRCULAR
PATH ANGLE: 0)*

WOULD BE MAPPED INTO ONE OF:

GPPE MANUFACTURING FEATURE:
1) SQUARE-U CIRCULAR FACE GROOVE
2) CIRCULAR FACE WITH TWO RESTRICTED SIDES

Figure 3-10 Sample Feature Mapping

1) Determine the **pi-classification** of the form feature: A pi-classification is essentially a text string which points to a series of ICAD defparts which define the rules for interpreting and extracting data for all elements in a particular pi-classification set. In other words, a pi-classification places similar form features into a group which may be processed according to a set of rules which apply specifically to that group.

The pi-classification is derived from the various components of each PDES Design Model element. In the case of a form feature, the pi-classification would be derived by applying a simple rule which combines path, profile, and feature types to arrive at the final pi-classification string. For example: a groove might have a "circular" path, a "u" profile, and a feature type of "depression." This feature's pi -classification would therefore become "u-circular-depression." There would therefore exist a rule set which would further interpret all "u-circular-depression" cases. In the case of other non-feature elements where no feature mapping need occur, the pi-classification would generally be derived from the PDES classification itself.

2) Determine the **mfg-type** of the given element: Each element's pi-classification identifies a rule-set which serves to further classify the feature and to define its appropriate values. One important attribute which is computed for each feature is the **mfg-type**. The mfg-type attribute represents the final PDES Interpreter interpretation of the element. In the case of the groove above, the mfg-type might be "u-circular-od-groove." The mfg-type is a key attribute which determines all future processing of each element in the PDES Interpreter. Determining the mfg-type can be a simple one-to-one mapping, as in the case of a cylindrical face or a marking. However, in some cases determining a mfg-type can be quite complex, requiring some extensive analysis of element properties.

3) Extract and calculate all appropriate attribute data for the element: Each material, face, feature, datum, tolerance and note of the PDES Interpreter must be able to respond to standard queries about its own attributes. The final step in the PDES mapping process is the instantiation of all necessary attribute values for each PDES Interpreter element. Each element of the PDES Interpreter conforms to a protocol which simplifies all downstream data requirements; for example, when demanded, a circular-path element would be expected to provide all geometric information including its diameter, orientation, location, and tolerance.

For a detailed look at the application of this three step methodology in the mapping process refer to 3.3.3.2.2.4 "**Features.**"

3.3.3.2.2 PDES Interpreter Components

The PDES Interpreter consists of nine major branches. Each branch represents a different object type (see Figure 3-11). The PDES Interpreter collects feature instances and other data from the PDES Design Model and segments this data into the various categories. The PDES Interpreter assures that each element conforms to the appropriate pr ocol for use by downstream units. The following sections detail

each of these categories and describe the methods used to perform the mapping from a PDES form-feature format to manufacturing features required for process planning.

```
Pdes-Interpreter
   Material
      Property 1
   Shape
   Faces
      (1580) Circular-Face -> Circular-Face
      (3460) Cylindrical-Face -> Outer-Diameter-With-No-Restricted-Sides
      (980) Circular-Face -> Circular-Face
   Features
      (4580) Square-U-Profile-Complete-Circular-Path-Depression -> Square-Circular
      (4880) Edge-Flat-Set
         (4760) Edge-Flat-3 -> Circular-Od-Edge-Flat
         (4740) Edge-Flat-3 -> Circular-Od-Edge-Flat
      (4960) Marking
      (5040) Marking
   Datums
      (4300) Plane-Datum -> Plane-Datum
   Dimensional-Tolerances
      (3860) Location-Dimension
      (3980) Size-Dimension
      (4680) Location-Dimension -> Location-Tolerance
      (4720) Location-Dimension -> Location-Tolerance
   Geometric-Tolerances
   Feature-Notes
      (3580) Feature-Machining-Allowed-Surface-Finish -> Feature-Surface-Finish
      (3700) Feature-Machining-Allowed-Surface-Finish -> Feature-Surface-Finish
      (4940) Feature-Marking -> Feature-Marking
      (5020) Feature-Marking -> Feature-Marking
   General-Notes
      (6120) General-Note -> General-Note
      (6180) General-Plating -> General-Plating
      (6220) General-Heat-Treat -> General-Heat-Treat
      (6260) General-Inspection -> General-Inspection
      (6300) General-Note -> General-Note
      (6360) General-Note -> General-Note
      (6380) General-Note -> General-Note
```

Figure 3-11 The PDES Interpreter

3.3.3.2.2.1 Materials

The materials component of the PDES Interpreter contains all material specifications as described in the original PDES data. PDES material properties are collected from the PDES Design Model utilizing the method described in 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements", and the materials component of the PDES Interpreter is instantiated.

3.3.3.2.2.2 Shape

The shape component consists of basic shape information culled from the PDES Design Model. The basic shape is the primary solid from which all other implicit features are subtracted. The primary

purpose of the shape function is to determine the basic shape type, and to gather all pertinent size, orientation, and element information about the particular basic shape. There are two basic shapes currently recognized bv the system. These are "cylindrical" and "prismatic." The basic shape type is determined by the following rule:

IF number of faces in the basic shape = 3
    THEN shape-type = 'cylindrical'
ELSE IF number of faces in the basic shape = 6
    THEN shape-type ='prismatic'
ELSE shape is not recognized

3.3.3.2.2.3 Faces

The face structure is instantiated with the face data from the PDES Design Model. The primary purpose of the face functions is to collect the part's basic shape faces and face data.

3.3.3.2.2.3.1. PDES Face Mapping

The mapping of PDES faces to manufacturing faces is accomplished using the same methods used for PDES feature interpretation (see 3.3.3.2.2.4 "Features").

The pi-classification for each face is determined by the following rule:

IF number of edges = 2
THEN pi-classification = CIRCULAR-FACE
ELSE IF number of edges = 4
    THEN pi-classification = RECTANGULAR-FACE
ELSE IF number of edges = 6
    THEN pi-classification = CYLINDRICAL-FACE
ELSE the face is not classifiable.

The mfg-type is determined in the following manner:

mfg-type = pi-classification

3.3.3.2.2.4 Features

The features component of the PDES Interpreter performs a form feature to manufacturing feature mapping and instantiates the structure with the feature data from the PDES Design Model. This feature mapping is accomplished utilizing the method described (see 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements").

For a complete detailed listing of all features which are supported in the GPPE 1.5 environment along with a listing of known feature support limitations, see 3.3.3.3 "Feature Support in GPPE V1.5."

The following section contains a detailed description of a feature mapping instance within GPPE. The general methodology applied in this example applies to all other major components of the PDES Interpreter.

3.3.3.2.2.4.1 PDES Feature Mapping Example

Each subtractive PDES feature volume, by definition, consists of a profile, a path, and a subtractive feature type. For example, a shoulder on a prismatic part would consist of a "vee" profile type, a "linear" path type, and a "depression" subtractive feature type.

The profile, path, and subtractive feature type are combined into the "pi-classification" which is used to evoke an ICAD defpart which defines the rules and protocols for a particular pi-classification set.

Example: For a square-u outer diameter groove, the following process would take place:

1) Determine the PDES feature's pi-classification

pi-classification for a feature = volume-classification+feature-type

Where volume-classification is defined on a feature-by-feature basis.

For a groove:

volume-classification = profile-classification + path-classification

Therefore, for a square-u groove:

pi-classification = SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION

2) Determine a mfg-type

Apply the rules identified by the pi-classification to further classify the feature, assigning it a "mfg-type."

The pi-classification is used to evoke a defpart of the same name (that is, SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION), which contains rules which further classify the feature, assigning it a "mfg-type."

For circular grooves, the assignment of a mfg-type is accomplished using the following rule:

```
IF the path is symmetric about the part z axis
AND
    IF the orientation angle = is away from the axis
        THEN mfg-type = SQUARE-CIRCULAR-OD-GROOVE
    ELSE IF the orientation angle = is parallel to the axis
        THEN mfg-type = SQUARE-CIRCULAR-FACE-GROOVE
    ELSE IF the orientation angle = is toward the axis
        THEN mfg-type = SQUARE-CIRCULAR-ID-GROOVE
    endif;
ELSE the feature is not interpretable
endif;
```

3) Extract and compute all attribute values required to meet the feature protocol requirements for downstream applications. (For a sample listing of all attribute values for a square-circular-od-groove feature instance, refer to Appendix VI)

3.3.3.2.2.5 Datums

The datum structure is populated with the datum data collected from the PDES Design Model utilizing the method described in 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements." The primary purpose of the datum functions is to collect all datums and their pertinent attribute values.

The pi-classification is determined very simply in the following manner:

pi-classification = the PDES datum reference classification.

The mfg-type is determined in the following manner:

mfg-type = pi-classification

3.3.3.2.2.5.1 Supported Datum Types

The supported datum types (those datum types which may be processed by the PDES Interpreter) include: geometric, plane, axis and feature.

3.3.3.2.2.6 Dimensional Tolerances

The dimensional tolerance structure is instantiated with the dimension data from the PDES Design Model utilizing the method described in 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements." The primary purpose of the dimension functions is to collect all dimensional tolerances and their pertinent attribute values.

The pi-classification is determined in the following manner:

pi-classification = the PDES tolerance classification.

The mfg-type is determined in the following manner:

```
IF pi-classification = "angle-dimension"
        THEN mfg-type = ANGLE-TOLERANCE
ELSE IF pi-classification = "size-dimension"
                AND the dimensioned entity is not a base-shape circular face
                THEN mfg-type = SIZE-TOLERANCE
        ELSE mfg-type = nil
ELSE IF pi-classification = "location-dimension"
                AND the dimensioned entity is not a base-shape circular face
                THEN mfg-type = LOCATION-TOLERANCE
        ELSE mfg-type = nil
ELSE the tolerance is not classifiable (nil)
```

3.3.3.2.2.6.1 Supported Dimensional Tolerance Types

The supported tolerance types (those dimensional tolerance types which may be processed by the PDES Interpreter) include: angle, size, and location.

3.3.3.2.2.7 Geometric Tolerances

The geometric tolerance structure is instantiated with the geometric tolerance data from the PDES Design Model utilizing the method described in 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements." The primary purpose of the geometric tolerance functions is to collect all geometric tolerances and their associated attribute values.

The pi-classification is determined in the following manner:

pi-classification = the PDES tolerance classification.

The mfg-type is determined in the following manner:

mfg-type = pi-classification

3.3.3.2.2.7.1 Supported Geometric Tolerance Types

The supported geometric tolerance types (geometric tolerance types which is processed by the PDES Interpreter) include: circularity, concentricity, cylindricity, parallelism, perpendicularity and position.

3.3.3.2.2.8 Feature Notes

The feature notes structure is instantiated with various types of property and feature data from the PDES Design Model utilizing the method described in 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements." Feature notes are annotations which refer to the properties of a specific feature. Examples of feature notes would include surface finishes and inspections, where a property such as a surface finish or inspection instruction refers to a specific feature of the part. Feature notes are mapped and built from many types of PDES entities. The primary purpose of the feature notes functions is to collect and represent those features, annotations and others which have been designated as being feature notes. A key function of the feature notes component, is that of formulating man-readable notes from non-man-readable feature or property data.

The pi-classification for each type of feature note is determined differently depending on the originating element. In general:

pi-classification = the PDES annotation classification
or a similar text string.
for example, **surface-texture** becomes **surface-finish**.

The mfg-type is determined in the following manner:

mfg-type = the pi-classification or some reasonable terminology modification
for example, **Feature-machining-allowed-surface-finish** becomes
Feature-surface-finish.

3.3.3.2.2.8.1 Supported Feature Note Types

Three types of feature notes are supported:

1) Chemical Properties, which include: coating, plating, and painting.
2) Inspection Properties, which include inspection.
3) Mechanical Properties, which include: markings, surface finish, and edge breaks.

3.3.3.2.2.8.2 Use of Feature Notes in the PDES Interpreter

Feature notes are sometimes used as a "catch-all" when it is unclear how an entity should be processed otherwise. An example of this would be markings. Markings represent a feature for which a process is required, and thus should be treated as a manufacturing feature. However, since no current capability for dealing with markings in a process plan is available, they are treated within GPPE as a special type of feature note.

3.3.3.2.2.9 General Notes

A general note is any note which is not associated with a specific feature. The general note structure is instantiated with all general note data from the PDES Design Model using the method described in 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements." The primary purpose of the general note functions is to collect general notes and their associated attribute values. As is the case with feature-notes, a key function of the general notes component, is formulating man-readable notes from non-man-readable

specifications. For example, a general inspection note might contain a string and a number of mil-specs. This must be formed in a readable format suitable for placement on an engineering drawing.

The pi-classification is determined in the following manner:

pi-classification = the PDES notes and specs classification.

The mfg-type is determined simply in the following manner:

mfg-type = pi-classification

3.3.3.2.2.9.1 Supported General Note Types

The supported general note types are similar to the feature note types. There are three basic general note types:

1) Chemical which includes: coating, plating, painting, penetrant inspection, other (for other chemical properties).
2) Inspection, which includes general inspection.
3) Other, which includes heat treat, marking, and a general note which may contain any type of free-form text.

3.3.3.3 Feature Support in GPPE V1.5

3.3.3.3.1 Levels of Feature Support within GPPE

GPPE supports feature processing at two levels, "bypass" and "generative." The system is capable of ascertaining whether a part being processed lies within GPPE's generative capability. If GPPE determines that the part is outside its capability, then bypass mode processing is recommended to the user.

3.3.3.3.1.1 Bypass Mode

Bypass mode is used when a specific manufacturing feature in a PDES file is not generatively supported by the GPPE. In bypass mode, GPPE bypasses the generative process planning functions and acts solely as a PDES importer. In this mode, the PDES data is imported and the PDES Design Model is instantiated, the PDES features are mapped into manufacturing features by the PDES Interpreter, and the Manufacturing Part Model is instantiated. The Manufacturing Part Model fulfills all PIF and Solid Model protocol requirements. Thus, in bypass mode, the system can generate a PIF, an ICAD solid model of the part, an IGES model of the part and, if desired, a Generic Requisition. When operating in bypass mode, the product information is transferred downstream for exception process planning by MEXP.

3.3.3.3.1.2 Generative mode

In generative mode, GPPE provides all capabilities of bypass mode, plus full generative capabilities. That is, capable of selecting processes for the given feature set, and generating a routing.

3.3.3.3.2 Detailed Feature Support

3.3.3.3.2.1 Features Supported in GPPE V1.5

Table 3-1 contains a listing of support levels for all manufacturing features within GPPE V1.5. All features in the table are supported in "Bypass" mode (see 3.3.3.3.1.1 "Bypass Mode"). A "yes" in the

"Generative Support" column indicates that the feature is also supported in "Generative Mode" (see 3.3.3.3.1.2 "Generative M... "). Refer to subsequent sections for actual feature mapping details.

Table 3-1 Feature Support Level within GPPE V1.5

Manufacturing Feature	Generative Support?
circular-face	Yes
rectangular-face	No
hex-face-pattern	No
outer-diameter-with-no-restricted-sides	Yes
outer-diameter-with-1-restricted-side	No
tapered-outer-diameter-with-no-restricted-sides	No
blind-hole	Yes
blind-inner-diameter	Yes
through-hole	Yes
through-inner-diameter	Yes
blind-tapered-inner-diameter	No
through-hex-hole	No
square-linear-groove-with-2-open-ends	No
square-circular-od-groove	Yes
square-circular-face-groove	No
u-linear-groove-with-2-open-ends	No
u-circular-od-groove	No
internal-thread-blind	No
internal-thread-through	No
external-thread	Yes
surface-flat-with-3-open-sides	No
linear-edge-round	No
circular-od-edge-round	No
circular-id-edge-round	No
circular-id-edge-flat	Yes
circular-od-edge-flat	Yes
diamond-knurl	No
edge-break	No
marking	No

3.3.3.3.2.1.1 PDES to Manufacturing Feature Mapping

Table 3-2 contains a listing of PDES features and the possible manufacturing feature mapping equivalents for those features supported within GPPE V1.5. Note that the table consists of both one-to-many relationships where a single PDES feature maps to multiple manufacturing features, and many-to-one relationships in which multiple PDES features map to a single manufacturing feature.

Table 3-2 PDES Features And Their Manufacturing Part Equivalents

PDES Features	Manufacturing Features
Circular Face derived from Basic Shape	circular-face
rectangular-face derived from basic shape	rectangular-face
cylindrical-face derived from the basic shape	outer-diameter-with-no-restricted-sides
Area/threads	external-thread
	internal-thread-through
	internal-thread-blind
Depression Circular/Square-u	square-circular-face-groove
	square-circular-od-groove
Depression Circular/Rounded-U	u-circular-od-groove
Depression Circular/V-Sweep	outer-diameter-with-1-restricted-side
Depression Linear/Square-u	square-linear-groove-with-2-open-ends
Depression Linear/Rounded-u	u-linear-groove-with-2-open-ends
Depression Linear/V-Sweep	surface-flat-with-3-open-sides
Depression Linear/N-Gon Profile 6, External	hex-face-pattern
Passage Linear/N-Gon Profile 6, Interior	through-hex-hole
Depression Linear/Constant Diameter	blind-hole
	blind-inner-diameter
Passage/Constant Diameter, Linear path	through-inner-diameter
	through-hole
Depression Linear/Tapered Diameter	blind-tapered-inner-diameter
	circular-id-edge-flat
Transition/Edge Flat	circular-id-edge-flat
	circular-od-edge-flat
	linear-edge-flat
transition/edge-round	circular-id-edge-round
	circular-od-edge-round
	linear-edge-round
Area/knurls,diamond	diamond-knurl
transition/edge break	edge-break
Area/Markings	markings

3.3.3.3.2.2 Feature Support Limitations and Assumptions

3.3.3.3.2.2.1 Known General Limitations

There are some known limitations to the current feature-mapping method in use by GPPE V1.5. These limitations are addressed here at two levels: first, the limitations of the feature-mapping algorithm itself will be discussed, and finally, some specific known limitations will be addressed.

3.3.3.3.2.2.1.1 Limitations of Product Model Interpretation Using the Feature Mapping Method

In the manufacturing domain, process planning involves the selection and sequencing of operations and processes necessary to transform raw material into a finished part. A feature decomposition for one particular part in essence comprises a very high-level process plan for the manufacture of that part. Thus the act of classifying a part region as being of one type of feature or another is itself, an extremely critical process planning action. To a large extent, the success of a process planning system depends on the ease with which it can classify feature information given any set of product data.

Interpretation of product data into features pertinent to a given manufacturing process is a complex problem. A one-to-one mapping between design and manufacturing features works in a limited sense, but often does not scale to more complex parts and manufacturing environments. Depending on the manufacturing domain, the machines, accepted manufacturing practice, tolerances, fixturing requirements and others, the desired feature decomposition of a given product will vary. Extensive process and domain knowledge is required to perform a good feature decomposition which makes sense for the given domain. The feature mapping method being employed by GPPE V1.5 currently does not, for the most part, take these constraints into account. Consequently, the resulting feature decompositions may not be the most desirable.

The following three paragraphs delineate more clearly some of the shortcomings of the feature mapping method currently in use.

A. Incoming Part Data Modeled in Terms of Manufacturing Features

In recent years it has been noticed that under certain conditions product design features seem to map directly to process design features. Thus was born the idea that features could serve as a major vehicle by which concurrent product and process design may be realized. More specifically, it has been suggested that those features used to design a product could also be used to drive the manufacture of that product. Unfortunately, such claims have been the result of limited academic success stories which do not necessarily scale to more complex parts and manufacturing environments. Direct translation between design and manufacturing features is very difficult; several design features (or portions thereof) may contribute to a single manufacturing feature, and vice versa. In fact, due to fixturing constraints or other variables, manufacturing features may often consist of geometry which is not a component of the as-designed part. In general, it is not feasible to require a designer to create a product using features and at the same time select the set of features which most adequately characterize the manufacture of the product.

B. Context Dependency of Features is Ignored

Given a part of at least moderate complexity, a designer and a manufacturer will provide very distinct views on the important features of that part. In addition, feature decomposition varies greatly within individual manufacturing domains. For example: machining features would consist of removal volumes, inspection features would consist primarily of various faces and profiles, while deburring features would consist of convex edges. The feature mapping method currently is capable of providing only a single decomposition, based on the machining features of the part.

C. Multiple Feature Alternatives are not Considered

For any machined part, there is virtually an infinite number of alternative feature decompositions. From this set, there is normally a relatively small number of decompositions that map to reasonable product or process designs. Each alternative decomposition can be mapped to a different process design. Consequently, selection of a "good" alternative involves detailed knowledge of the techniques and resources of a particular domain. GPPE's feature mapping algorithms arrive at only a single feature decomposition which must be utilized to generate the process plan.

3.3.3.3.2.2.1.2 GPPE V1.5 Feature Support Limitations

There are known feature support limitations within GPPE V1.5. Some are due to limitations in PDES, others are caused by GPPE V1.5 code limitations. The following sections contain a brief general listing of those known limitations.

A. Extent of Feature Testing

The GPPE V1.5 supported feature set (see Table 3-1) has been tested only with the PDES Prove-out Technical Data Package (TDP) set. GPPE feature support has been made as robust as possible, but there are both known and certainly unknown cases in which these features simply will not work due to unforeseen feature combinations, PDES limitations, or other complications. (For example, it is known that edge-flats and edge rounds will not work properly when placed at the intersection of a groove and another feature. This is due to the fact that the PDES implicit representation of grooves does not allow feature constituents to be individually referenced. Therefore, it is unclear where the edge treatment is to be placed).

B. Stock Removal Errors

GPPE V1.5 does not implement solid Work-in-process capability. Solid Work-in-process capability uses the solid model to determine remaining work material. Problems can occur if proper consideration of previously removed material is not taken. Such problems might include: overestimated cutting times, and MetCAPP/tool-selection errors (for example, without solid Work-in-process capabilities, a groove may be defined which is so deep that it appears to be impossible to machine when in reality, the groove may be very shallow once the other features above it are removed).

C. Implicit Feature Limitations

Form features are currently represented implicitly within PDES. With implicit feature representations, it is not possible to reference individual geometric elements of a feature. This can result in three known difficulties:

1) Tolerance Ambiguities: Location tolerance specifications may be ambiguous. In the example (see Figure 3-12), a groove is located with respect to a blind inner diameter. The PDES definition states simply that the location tolerance between the groove and the hole is ± .xx. Within the context of this simple tolerance specification, it is not clear to which of the six cases the tolerance should be applied.

2) Transition Feature Ambiguities: The application of transition features may be ambiguous. Because individual implicit feature elements are not addressable, the location and orientation of a transition feature may be unclear. For example, if an edge flat is to be located at the intersection of a groove and an outer diameter, it is unclear to which leg of the groove profile the transition feature should be attached.

3) Solid Modeler Errors: Because different solid modelers vary in their precision, it is possible that an implicit form feature may be over or underspecified in the PDES model, resulting in a non-manifold solid, either within GPPE, or in the downstream CAD/CAM system. In the current GPPE configuration, this has not yet been a problem.

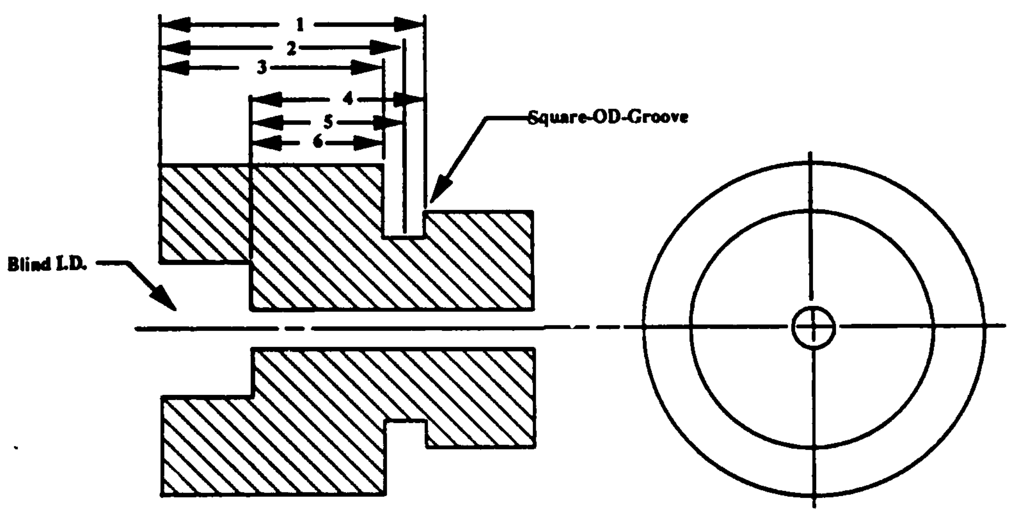


Figure 3-12 Implicit Feature Representation Leads to Tolerancing Ambiguities

D. MetCAPP Feature Support Limitations

1) Limited MetCAPP Feature Set Support: MetCAPP supports fewer than forty different processes (for a complete listing of supported MetCAPP processes refer to the "MetCAPP User's Guide"). Some of the features which are currently supported in bypass mode, are not supported by any MetCAPP process and must be supported through the development of a custom manufacturing rule-set to determine appropriate operations, to select tools, and to calculate passes, feeds, speeds, and other data.

2) MetCAPP Geometry Limitations: MetCAPP defines geometric minimums and maximums for each of its processes. For example, for the MetCAPP groove process, feeds and speeds are not supplied for grooves of less than .255 inches in diameter. If the GPPE manufacturing feature diameter is smaller than this minimum, the feature requires non-MetCAPP support.

3.3.3.3.2.2.2 Detailed Known Feature Limitations

In the course of GPPE system development, feature support has been demonstrated and approved for each of the features in Table 3-1. In this feature development and refinement process, several specific limitations for specific cases of individual features have been noted. Some of these limitations are due to PDES data deficiencies, others are due to MetCAPP deficiencies, and some may be attributed to the feature mapping methodology currently in use. Certain limitations are due to the fact that a specific functionality has simply not yet been incorporated into the system. The following sections detail such known limitations, and in addition, specify some feature assumptions.

3.3.3.3.2.2.2.1 Basic shape

If the basic shape is prismatic, then there is no consideration of "turned" features such as circular grooves or inner diameters. In reality it is possible to have a mostly prismatic part with features that can be turned, and, conversely, a mostly axisymmetric part with features that cannot be turned.

All manufacturing features are, by definition, subtracted from the basic shape. In other words, the basic shape currently serves as the starting block or cylinder from which the features are "carved out." Representing features which are protrusions from the basic shape can only be accomplished by enlarging the basic shape and subtracting the material around the protrusion.

3.3.3.3.2.2.2.2 Circular Face

Limitations for this feature are the same as those which apply to the basic shape. (See 3.3.3.3.2.2.2.1 "Basic Shape.")

3.3.3.3.2.2.2.3 Rectangular Face

Limitations for this feature are the same as those which apply to the basic shape. (See 3.3.3.3.2.2.2.1 "Basic Shape.")

3.3.3.3.2.2.2.4 Hex-Face

The PDES definition currently in use for this feature is inappropriate . This feature is currently identified in PDES as an implicit depression, and uses an in-out feature sweep with a specified removal direction to declare that it is actually a face rather than a hole. In-out feature sweeps should normally only be applied to material removal operations. Upon implementation of feature pattern capabilities, this feature will be represented correctly.

3.3.3.3.2.2.2.5 OD with No Restricted Sides

Limitations for this feature are the same as those which apply to the basic shape. (See 3.3.3.3.2.2.2.1 "Basic Shape.")

3.3.3.3.2.2.2.6 OD with One Restricted Side

Feature depth calculation due to lack of solid work-in-process capability is an issue. (See 3.3.3.3.2.2.1.2 paragraph B. "Stock Removal Errors.")

This feature may be prone to implicit feature representation ambiguity difficulties (see 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations").

It is assumed that this feature is formed by a 90 degree "L" shaped profile, with a rotational axis which is parallel to the turned axis of the part.

It is assumed that location tolerances refer to the vertical leg of the "L" profile.

3.3.3.3.2.2.2.7 Tapered Outer Diameter with no restricted sides

Limitations for this feature are the same as those which apply to the circular OD edge flat. (See 3.3.3.3.2.2.2.26 "Circular OD Edge Flat.")

Acceptable PDES test data for this feature is not currently available; RPGS limitations do not currently support the Depression Linear/Tapered Diameter feature.

The distinction between a tapered outer diameter and an OD edge flat is not well-defined, and is therefore left to the user to resolve at the process level.

3.3.3.3.2.2.2.8 Blind Hole (Flat End or Conical End)

Without explicit geometry reference capability (see 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations"), hole depth tolerances may be ambiguous in some cases.

It is assumed that, in the case of a blind hole with a conical end, the tolerance relating to hole depth is associated with the distance from the hole origin to the conical-end tip.

See 3.3.3.3.2.2.2.10 "Through Hole" for clarification of the distinction between holes and inner diameters.

3.3.3.3.2.2.2.9 Blind Inner Diameter

Limitations are the same as those which apply to the blind hole. (See 3.3.3.3.2.2.2.8 "Blind Hole").

3.3.3.3.2.2.2.10 Through Hole

Through holes are defined in PDES as having a length. Although this results in a correct solid, without solid work-in-process capability, having a through hole which is defined as being longer than the

material which it goes through can give false operation times and errant tool constraints for the manufacture of the feature. In addition, this hole length problem can cause difficulty with any area features such as threads, which are attached to the through hole feature. (See 3.3.3.3.2.2.1.2 paragraph B. "Stock Removal Errors.")

NOTE: A hole (as opposed to an inner diameter) by definition, is any axisymmetric depression or passage which is **not** oriented along the main axis of a turned part. This implies that "holes" may be manufactured only by "holemaking" operations. An inner diameter is any axisymmetric depression or passage which **is** oriented along the main axis of a turned part. An inner diameter may be manufactured either by a "turning" or a "holemaking" operation.

3.3.3.3.2.2.2.11 Through Inner Diameter

Limitations for this feature are the same as those which apply to the through hole. (See 3.3.3.3.2.2.2.10 "Through Hole.")

3.3.3.3.2.2.2.12 Blind Tapered Inner Diameter

The rules which determine whether the same PDES form feature is mapped to a blind tapered inner diameter or an Inner diameter edge flat are somewhat arbitrary, and are resolved by the user at the process level.

3.3.3.3.2.2.2.13 Through Hex Hole

There is currently no Metcut process capable of producing this feature. (See 3.3.3.3.2.2.1.2 paragraph D. "MetCAPP Feature Support Limitations.")

3.3.3.3.2.2.2.14 Square Linear Groove with Two Open Ends

This feature may be prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

A Square groove in which fillet-1 + fillet-2 >= groove width is not supported.

Groove profile orientations which are not normal to the surface will have unpredictable results.

Groove depth calculation due to lack of solid work-in-process capability may be in error. (See 3.3.3.3.2.2.1.2 paragraph B. "Stock Removal Errors.")

It is assumed that all groove location tolerances refer to the groove profile center.

3.3.3.3.2.2.2.15 Square Circular OD Groove

This feature may be prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

A square groove in which fillet-1 + fillet-2 >= groove width is not supported.

Groove depth calculation due to lack of work-in-process capability may be in error. (See paragraph 3.3.3.3.2.2.1.2 B. "Stock Removal Errors.")

It is assumed that the groove prof is normal to the outer surface

It is assumed that all circular grooves lie on the axis of the turned part.

It is assumed that all groove location tolerances refer to the groove profile center.

3.3.3.3.2.2.2.16 Square Circular Face Groove

The limitations which apply to this feature are the same as those which apply to square grooves. (See 3.3.3.3.2.2.2.15 "Square Circular OD Groove.")

Face grooves are not currently supported by MetCAPP. (See 3.3.3.3.2.2.1.2 paragraph D. "MetCAPP Feature Support Limitations.")

3.3.3.3.2.2.2.17 U Linear Face Groove with Two Open Ends

This feature may be prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Groove depth calculation due to lack of work-in-process capability is an issue. (See 3.3.3.3.2.2.1.2 paragraph B. "Stock Removal Errors.")

It is assumed that all groove location tolerances refer to the groove profile center.

3.3.3.3.2.2.2.18 U Circular OD Groove

This feature may be prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Groove depth calculation due to lack of solid work-in-process capability is an issue. (See 3.3.3.3.2.2.1.2 paragraph B. "Stock Removal Errors.")

It is assumed that all circular grooves lie on the axis of the turned part.

It is assumed that all groove location tolerances refer to the groove profile center.

3.3.3.3.2.2.2.19 Internal Thread/Tap (blind and through)

Thread length may be excessive due to PDES hole feature representation. This may result in an incorrect solid, and excessive cutting times. (See 3.3.3.3.2.2.2.8 "Blind Hole" and 3.3.3.3.2.2.2.10 "Through Hole.")

3.3.3.3.2.2.2.20 External Thread

Thread milling is currently not supported by MetCAPP. (See 3.3.3.3.2.2.1.2 paragraph D. "MetCAPP Feature Support Limitations.")

3.3.3.3.2.2.2.21 Surface Flat with Three Open Sides

Feature depth calculation due to lack of solid work-in-process capability is an issue. (See 3.3.3.3.2.2.1.2 paragraph B. "Stock Removal Errors.")

3.3.3.3.2.2.2.22 Linear Edge Round

This feature is supported as a transition only between the following combinations of PDES features

1) Rectangular Face and Rectangular Face
2) Rectangular Face and Depression Linear/V-Sweep

Feature combinations which fall outside of these two cases are not supported in generative or bypass modes.

Because this feature is a transition feature, it may be particularly prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Shoulders on rounds are assumed to be 90 degrees. Other angles may lead to incorrect solid representation.

3.3.3.3.2.2.2.23 Circular OD Edge Round

This feature is supported as a transition only between the following combinations of PDES features

1) Circular Face and Cylindrical Face
2) Circular Face and Depression Circular/V-Sweep
3) Cylindrical Face and Depression Circular/V-Sweep
4) Depression Circular/V-Sweep and Depression Circular/V-Sweep

(all "Depression Circular/V-Sweep's" are assumed at this time to conform to "OD with One Restricted Side" limitations. (See 3.3.3.3.2.2.2.6 "OD with One Restricted Side.")

Feature combinations which fall outside of these four cases are not supported in generative or bypass modes.

Because this feature is a transition feature, it may be particularly prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Shoulders on rounds are assumed to be 90 degrees. Other angles may lead to incorrect solid representation.

3.3.3.3.2.2.2.24 Circular ID Edge round

This feature is supported only as a transition between the following combinations of PDES features

1) Circular Face and Passage/Constant Diameter
2) Circular Face and Depression/Constant Diameter
3) Rectangular Face and Depression/Constant Diameter

Feature combinations which fall outside of these three cases are not supported in generative or bypass modes.

Because this feature is a transition feature, it may be particularly prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Shoulders on rounds are assumed to be 90 degrees. Other angles may lead to incorrect solid representation.

3.3.3.3.2.2.2.25 Circular ID Edge Flat

This feature is supported only as a transition between the following combinations of PDES features:

1) Circular Face and Passage/Constant Diameter
2) Circular Face and Depression/Constant Diameter
3) Rectangular Face and Depression/Constant Diameter

Feature combinations which fall outside of these three cases are not supported in generative or bypass modes.

Because this feature is a transition feature, it may be particularly prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Shoulders on flats are assumed to be 90 degrees. Other angles may lead to incorrect solid representation.

3.3.3.3.2.2.2.26 Circular OD Edge Flat

This feature is supported only as a transition between the following combinations of PDES features

1) Circular Face and Cylindrical Face
2) Circular Face and Depression Circular/V-Sweep
3) Cylindrical Face and Depression Circular/V-Sweep
4) Depression Circular/V-Sweep and Depression Circular/V-Sweep

(It is assumed that V-Sweeps conform to OD with one restricted side limitations. (See 3.3.3.3.2.2.2.6 "OD with One Restricted Side.")

Feature combinations which fall outside of these three cases are not supported in generative or bypass modes.

Because this feature is a transition feature, it may be particularly prone to implicit feature representation ambiguity difficulties. (See 3.3.3.3.2.2.1.2 paragraph C. "Implicit Feature Limitations.")

Shoulders on flats are assumed to be 90 degrees. Other angles may lead to incorrect solid representation.

3.3.3.3.2.2.2.27 Diamond Knurl

Knurls are not currently supported by MetCAPP. (See 3.3.3.3.2.2.1.2 paragraph D. "MetCAPP Feature Support Limitations.")

3.3.3.3.2.2.2.28 Edge Break

This feature is not currently explicitly supported by MetCAPP. (See 3.3.3.3.2.2.1.2 paragraph D. "MetCAPP Feature Support Limitations.")

This feature is mapped by the PDES Interpreter to a feature note representation.

3.3.3.3.2.2.2.29 Marking

Markings are currently mapped by the PDES Interpreter into feature notes rather than into a manufacturing feature.

Markings are not currently supported by MetCAPP. (See 3.3.3.3.2.2.1.2 paragraph D. "MetCAPP Feature Support Limitations.")

3.3.3.4 PDES Interpreter Outputs

The PDES Interpreter has no real outputs in the standard sense, but exists simply as an internal representation of interpreted PDES data. The PDES Interpreter responds to requests for attribute values of the elements which it contains. (See 3.2.1.2.3 "The Demand-driven Nature of IDL.") Thus the outputs of the PDES Interpreter may be considered to be all information about the current PDES manufacturing feature mapping.

3.3.4 Manufacturing Part Model Summary

The Manufacturing Part Model is the final manufacturing feature-based part model on which all process planning operations are performed. The Manufacturing Part Model serves four primary functions.

1) Represent part features in their entirety as required by downstream process planning functions.
2) Generate solid and wireframe models for visualization purposes
3) Generate a PIF to be used in the creation of an engineering drawing on a Computer Aided Design (CAD) system.
4) Provide interactive capability for material selection.

3.3.4.1 Manufacturing Part Model Inputs

The Manufacturing Part Model utilizes the PDES Interpreter, the PDES Design Model, the Resource Model, and the Process model as data sources. Data is requested on an as-needed basis.

3.3.4.1.1 External Data

The Manufacturing Part Model makes use of the materials catalog. (This is discussed in detail in 3.3.4.2.4 "Raw Stock Selection.")

3.3.4.2 Manufacturing Part Model Processing

As is the case with the PDES Interpreter, each element in the Manufacturing Part Model subscribes to protocols and obeys given conventions (as described in 3.2.1.2.4 "Protocols"). The first protocol is that for the basic feature which allows the feature to interface to the information extracted from the PDES interpreter or some other data source. The second protocol provides appropriate graphics and interfaces the feature to the solid modeling system, and the third protocol interfaces the elements to a CAD system.

The Manufacturing Part Model consists of seven major divisions or branches (see Figure 3-13). These branches echo those in the PDES Interpreter, with the exception of "faces" which are represented in the Manufacturing Part Model as features, and raw stock, which is peculiar to the Manufacturing Part Model.

3.3.4.2.1 Manufacturing Part Model Instantiation

3.3.4.2.1.1 Feature Refinement

Based on the mfg-type defined for each feature in the PDES Interpreter (see 3.3.3.2.1.1 "Methodology for Mapping of PDES Elements"), a Manufacturing Part Model defpart of the same name is instantiated. This defpart contains rules which further refine each feature and assure that each feature conforms to the appropriate protocols.

3.3.4.2.1.1.1 Feature Sizing

The Manufacturing Part Model assures that each feature is fully specified, or dimensionally complete. Some form features are not dimensionally complete as they are represented in PDES; their dimensions are derived by the Manufacturing Part Model from their relationship to other features. After the PDES Interpreter has performed the mapping from PDES features to manufacturing features, the

```
Mfg-Part-Model
    Basic Shape:  Cylindrical [Diameter = 1.995", Length = 5.125"]
    Features
        (1) Circular Face At (0.0 5.125 0.0)
            Process 1:  Facing To Center
                Machine 1:  Small Turning Center
            Process 2:  Cut Off
                Machine 1:  Small Turning Center
        (2) Outer Diameter With No Restricted Sides At (0.0 2.562 0.0)
            Process 1:  Outer Diameter Turning Through
                Machine 1:  Small Turning Center
        (3) Circular Face At (0.0 0.0 0.0)
            Process 1:  Facing To Center
                Machine 1:  Small Turning Center
            Process 2:  Cut Off
                Machine 1:  Small Turning Center
        (4) Square Circular Od Groove At (0.0 2.57 0.0)
            Process 1:  Outer Diameter Grooving
                Machine 1:  Small Turning Center
        (5) Circular Od Edge Flat At (0.0 0.016 0.0)
            Process 1:  Form Turning
                Machine 1:  Small Turning Center
            Process 2:  Contour Turning
                Machine 1:  Small Turning Center
        (6) Circular Od Edge Flat At (0.0 5.109 0.0)
            Process 1:  Form Turning
                Machine 1:  Small Turning Center
            Process 2:  Contour Turning
                Machine 1:  Small Turning Center
    Datums
        (1) Plane Datum
    Tolerances
        (1) Location Tolerance
        (2) Location Tolerance
    Feature-Notes
        (1) Feature Surface Finish
        (2) Feature Surface Finish
        (3) Feature Marking
        (4) Feature Marking
    General-Notes
        (1) General Note
        (2) General Plating
        (3) General Heat Treat
        (4) General Inspection
        (5) General Note
        (6) General Note
        (7) General Note
    Raw-Stock
        ...
```

Figure 3-13 The Manufacturing Part Model

Manufacturing Part Model takes these manufacturing features and resizes them as necessary. For example, a square-u profile for a groove in PDES is represented as having infinite height. The Manufacturing Part Model representation of this feature contains rules which state t[illegible] a groove's height will be the distance from the profile base outward to the basic shape boundary. Othe[illegible]n rules apply to features such as through holes and outer diameters. These dimensions are automatically determined, assuring that the feature will conform to specific process planning protocols, solids generation protocols, and the PIF protocol.

3.3.4.2.1.1.2 Process and Machine Mapping

The Manufacturing Part Model provides the user with a view of a feature-to-process mapping for each feature, and further, lists each machine capable of realizing the feature (see Figure 3-13). For detail on this feature to process mapping refer to 3.3.6.3.1 "Process Model, Feature to Process Mapping", for further detail regarding machine selection, refer to 3.3.7.2.2.1 "Process Sequencer, Machine." The processes and machines which are displayed in the Manufacturing Part Model are for inspection only, and are not directly referenced by other GPPE components.

3.3.4.2.2 Solids and Wireframe Generation

GPPE builds a solid model of the part by instantiating the fully specified manufacturing features and performing boolean subtractions of each manufacturing feature from an appropriately sized basic shape. The resulting solid model may be rendered in either hidden-line-removed or "wireframe" modes, and may be viewed from any angle. This capability allows the user to visually verify that the incoming data was usable, and that the manufacturing part model resembled the part which was wanted.

Apart from the solid model, GPPE also builds wireframe representations of each feature and the basic shape. The wireframe model is used for quick feature visualization, and is used in conjunction with the solid model for feature identification (the user may select a "blink node" button which will overlay a blinking wireframe feature over the solid model).

The solid modeler and wireframe capabilities are provided through the ICAD Browser. (See 3.2.1 "The GPPE Implementation Environment.")

3.3.4.2.3 PDES Intermediate File (PIF)

3.3.4.2.3.1 Purpose

The PIF is a feature-based textual representation of the PDES model which serves as input to a program called CVTrans (a sample PIF is shown in Appendix III). CVTrans processes the PIF and generates a script which constructs and annotates a solid model in a target CAD environment. Currently CVTrans is targeted for the Computervision CADDS4X environment. In this mode, CVTrans generates a CADDS executable which builds an annotated solid model which is used for part visualization, Numerical Control (NC) toolpath generation, fixture design, and other downstream applications which rely on a fully annotated solid.

3.3.4.2.3.2 PIF Generation

The PIF is generated using a set of ICAD tools called **companions and writers**. (See 3.3.1.7 "Generating Text Output.") For each element in the Manufacturing Part Model, there exists a companion which extracts all appropriate values required for the PIF. For example: for each feature, a companion exists which extracts all dimensional information necessary for the PIF (this data would consist of basically the same data which was required to generate the ICAD solid), for each note, the note string would be extracted, for a thread specification, all thread data would be extracted. The writers in turn take the data and format it correctly.

3.3.4.2.3.3 IGES File Generation

An IGES 3.0 file may be generated from the solid which represents the nominal wireframe model of the current part instance (a sample IGES file is shown in Appendix VII). IGES generation capabilities are provided by the ICAD environment. (For a complete listing of supported IGES entities, refer to the "IGES CAD Input Tools user's Manual.")

3.3.4.2.4 Raw Stock Selection

Raw stock selection is the first component of GPPE which requires any user interaction. The Raw stock component of the Manufacturing Part Model consists of two major components: Material, and Stock Item.

3.3.4.2.4.1 Material

The Material component contains the material requirements as specified in the PDES source. This material data may be reviewed by the user when making a material selection. (See 3.3.1.3 "Customized Button Functions.")

3.3.4.2.4.2 Stock Item

Stock Item selects all materials from the in-house stock database which are appropriate given the basic shape and the material requirements as specified in the PDES source, and provides an interaction window which allows the user to select the raw stock from which the part will be machined. Stock Item obtains all material information from the Resource Model (see 3.3.5.2.2 "Materials").

3.3.4.2.4.2.1 Determining Valid Materials

Stock Item determines those in-stock materials which meet the specifications for the current part by adding a machining stock allowance value to the critical dimension of the basic shape, and querying the Resource Model for all materials which meet the dimensional requirements. The result of this query is a list of dimensionally valid materials. For a cylindrical part, the stock allowance is added to the diameter, for a prismatic part, the critical dimensions are multiple and the machining stock allowance is added to the basic shape's length width and height.

3.3.4.2.4.2.2 User Interaction

The user is presented with a list of valid materials and may select from the list, or may increase the range of the materials to be examined by increasing the machining stock allowance value. (See 3.3.1.4 "Specifying Input Values.")

3.3.4.2.4.2.3 Generic Requisition Generation

The Generic Requisition or BOMs (see Figure 3-14) is generated in much the same way as the PIF. (See 3.3.4.2.3.2 "PIF Generation.") A companion and writer set exists which appropriately formats the raw stock information into the appropriate format.

```
0000000400000201M000000000000291      3.5 dia rd. bar steel 3.107 in. long
```

Figure 3-14 Sample Generic Requisition

3.3.4.3 Manufacturing Part Model Outputs

The manufacturing part model has three tangible forms of output. These are the Generic Requisition, the PIF, and the IGES file. In addition, the Manufacturing Part Model, like the PDES Design Model and the PDES Interpreter exists itself as a data source for other modules. The Manufacturing Part Model answers requests for attributes of all elements which it contains. (See 3.2.1.2.3 "The Demand-driven Nature of IDL.") Thus, one of the outputs of the Manufacturing Part Model may be considered to be the Manufacturing Part Model itself or the set of all manufacturing features. Indeed, the manufacturing features which are represented in the Manufacturing Part Model form the basis for all subsequent process planning activity.

3.3.5 Resource Model Summary

The resource model collects data regarding available resources, and represents it internally in a format that is easily accessible by other modules (see Figure 3-15). The resource model is feature independent; regardless of the current part's manufacturing and material requirements, the Resource Model represents the full set of available resources. The resource model is used by other modules to determine available workstations, materials, and fixtures. Although "tools" appears in the Resource Model tree structure, this capability has not yet been developed.

3.3.5.1 Resource Model Data Sources

With the exception of fixturing data, all Resource Model input data is represented as ICAD catalogs. (See 3.2.1.4.1 "Catalogs.")

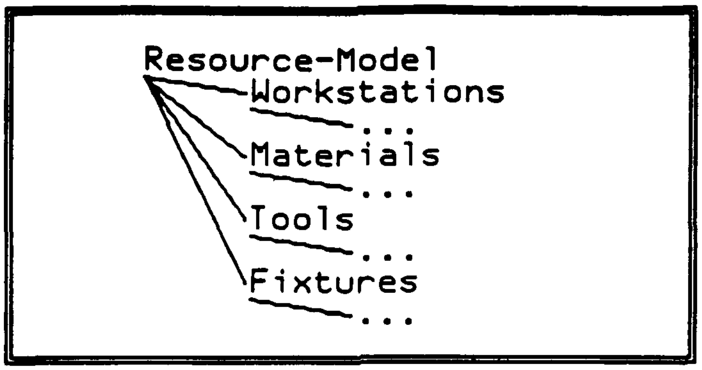


Figure 3-15 The Resource Model

3.3.5.1.1 Workstation Data

All machining and non-machining workstation data is stored in the "workstation-info" catalog. (For a sample workstation-info catalog refer to Appendix IV-A.)

3.3.5.1.2 Material Data

All material specifications are stored in the "materials" catalog. This catalog is generated from the ME database through a series of Oracle queries. (For a sample materials catalog refer to Appendix IV-B.)

3.3.5.1.3 Fixturing Data

Fixturing information is stored in the global variable "fixture-concepts-list." (For a sample fixture-concepts-list, refer to Appendix IV-E.)

3.3.5.2 Resource Model Instantiation

The resource model is populated with data from external ICAD catalogs and an internal global variable. The Resource Model consists of three major components: "workstations," "materials," and "fixtures." The fourth branch of the Resource Model, "tools" is not currently instantiated.

3.3.5.2.1 Workstations

All workstation information is extracted, segmented and internalized from the "workstation-info" table using the Relational Object Manager. (See 3.2.1.4.2 "Relational Object Manager.") Workstations are broken down into Machining and non-machining categories. The machining workstations are further broken down into a machine type of either milling, turning, or holemaking. (See Figure 3-16.) The resource model workstation data does not contain a full representation of each workstation and its characteristics. This detailed workstation information is available in the MetCAPP database. (See 3.3.6.2 "Process Model External Data.")

3.3.5.2.2 Materials

All material specifications are extracted, segmented and internalized using the Relational Object Manager. Material data is divided into two categories: cylindrical and prismatic. All information contained in the materials table is extracted, and is accessible by other modules (see Figure 3-17).

- Workstations
  - Machining-Workstations
    - Milling
      - Nichols Horizontal Hand Mill
      - Small Horizontal Machining Center
      - Hurco Cnc Bed Mill
    - Turning
      - Small Turning Center
      - Mazak Quick Turn 8
      - Logan Turret Lathe
      - Takisawa Ts-15
      - Star Swiss Lathe
      - Brown & Sharp Automatic 0g
      - Hardinge With Bar Feed
      - Brown & Sharp Automatic 00g
    - Holemaking
      - Acme Drill-Single Spindle
      - Small Horizontal Holemaking Center
      - Rockwell - Drill/Tap
  - Non-Machining-Workstations
    - Inspect Before Chem Film
    - Swayco
    - Inspect Before Anodize
    - Inspect From Matl Vendor
    - Kwik-Way Cylindrical Grinder
    - Roll-Abrator Tumbler
    - External Processing
    - Inspect Before Plate
    - Inspect From Comp Vendor
    - Final Inspection
    - Gyrock Mini-Tumbler
    - Inspect Before Paint
    - Bag & Seal
    - Peerless Automatic Bandsaw Setup
    - Bag, Seal And Bar Code
    - Inspect Before Heat Treat
    - Deburr

Figure 3-16 Sample Workstations

- Materials
  - Cylindrical
    - Steel 3.500 D, Aisi 1020, Astm A29, Qq-W-461, Hot Roll
    - Steel 0.750 D, 80-100 Rb, Aisi 1018, Cold Drn
    - Steel 1.125 D, Astm A108, Cold Drn, Matl Des Csuj
    - Aluminum 1.000 D, Astm B211, Qq-A-200/8, Cold Drn, Matl Des Albu, Temper T6511
    - Copper Alloy 3.500 D, Astm B505 932, Qq-C-390, Class C93200, Cond E7
    - Hidden
  - Prismatic
    - Steel 0.250 X 3.250, Aisi 1018, Astm A29, Mil-S-11310, Qq-W-461, Hot Roll
    - Aluminum 0.500 X 0.750, Astm B211, Qq-A-200/8, Cold Drn, Temper T6511
    - Titanium 0.375 X 5.250, Mil-T-9046, Hot Roll, Hot Fin
    - Steel 0.375 X 5.000, Aisi 1018, Astm A29, Mil-S-11310, Qq-W-461, Hot Roll
    - Steel 0.375 X 3.250, Aisi 1018, Astm A29, Mil-S-11310, Qq-W-461, Hot Roll
    - Steel 0.500 X 3.500, Aisi 1018, Astm A29, Mil-S-11310, Qq-W-461, Hot Roll
    - Aluminum 4.000 X 4.000, Qq-A-200/8, Cold Drn, Matl Des Albu, Temper T6511

Figure 3-17 Sample Materials

3.3.5.2.3 Fixtures

Fixture concept data is stored as a global variable. As such, this data is directly accessible, and is used to populate the fixture structure of the Resource Model (see Figure 3-18). The fixturing data includes information regarding the build-up of each fixture type; however, the capability to utilize this information has not yet been developed.

- Fixtures
  - Interior Chucking On Front Of Part
  - Interior Chucking On Rear Of Part
  - Exterior Chucking On Front Of Part
  - Exterior Chucking On Rear Of Part
  - Interior Chucking On Front Of Part
  - Interior Chucking On Rear Of Part
  - Exterior Chucking On Front Of Part
  - Exterior Chucking On Rear Of Part
  - Center Hole Dowel Location - 3 Pt Contact
  - Center Hole Dowel Location - 4 Pt Contact
  - Interior Chucking On Front Of Part
  - Interior Chucking On Rear Of Part
  - Exterior Chucking On Front Of Part
  - Exterior Chucking On Rear Of Part
  - Center Hole Dowel Location - 3 Pt Contact
  - Center Hole Dowel Location - 4 Pt Contact

Figure 3-18 Sample Fixturing Concepts

3.3.5.3 Resource Model Limitations and Assumptions

The Resource Model, combined with the populated MetCAPP database, provide all necessary resource information to the GPPE modules. Currently, the Resource Model accesses catalog representations of resource data rather than the Manufacturing Engineering database itself. Similarly, MetCAPP maintains its own internal resource representation. (See 3.3.6.1 "Process Knowledge Representation in the Process Model.") This manually maintained data redundancy complicates site-customization and inhibits the ability of the system to access current resource data.

Because MetCAPP relies on its own internal resource data, a cross-reference between Resource Model and MetCAPP resources must be maintained. (See 3.3.6.2.3.1 "Metcut Material Mapping.") Currently there is no dynamic method for defining the mapping between MetCAPP resource data and site-specific data.

3.3.5.4 Resource Model Outputs

The Resource Model has no real outputs in the standard sense, but exists as an internal representation of current resource data. The Resource Model answers requests for attributes of the elements which it contains. (See 3.2.1.2.3 "The Demand-driven Nature of IDL.") Thus, the outputs of the Resource Model may be considered to be all the information about the current resource state.

3.3.6 Process Model Summary

Within the Process Model, each generatively supported feature (see 3.3.3.3.2.1 "Features Supported in GPPE V1.5") is associated with a set of manufacturing processes that may be capable of producing it. Similarly, each potential manufacturing process is associated with one or more machining operations which may be required to create the particular instance of the feature, given its dimensions. Having accomplished the mapping from manufacturing features to processes and from processes to operations, the Process model is responsible for providing operation detail including speeds, feeds, tool selection, and most importantly, total operation times.

The Process Model tree is composed of three major divisions: Features, RAMP processes, and Metcut processes (see Figure 3-19). RAMP processes and Metcut processes branches exist for inspection purposes only, and are not used as data sources by any other GPPE functions. The features branch contains all data and functions which are necessary to perform a feature to process mapping and to compute all quantitative plan data.

3.3.6.1 Process Knowledge Representation in the Process Model

Computing realistic estimates of times required to manufacture a part cannot be accomplished without knowledge of required machining operations for each given feature, along with the number of tool passes required for each operation, and the speeds and feeds per pass. To compute this type of data, a huge amount of knowledge is required. This knowledge-base must include knowledge regarding the mapping from features to processes, knowledge of machines (their capabilities and limits), knowledge of generally accepted shop practices, material machinability data, tool selection rules, and so on. The

```
Process-Model
  Features
    ...
  Ramp-Processes
    ...
  Metcut-Processes
    ...
```

Figure 3-19 The Process Model

Process Model is the main repository for this manufacturing knowledge in GPPE. Representing this huge body of knowledge is accomplished by incorporating an interface to MetCAPP into the GPPE environment.

3.3.6.1.1 Metcut/MetCAPP

GPPE's sole source of manufacturing knowledge is currently that which is obtained from the Institute of Advanced Manufacturing Sciences (IAMS), Inc.'s **MetCAPP** product. (Note: Within the RAMP environment, the MetCAPP product has historically been referred to as "Metcut." For this reason, much of the GPPE V1.5 source code also refers to "Metcut." For the purpose of this document, and to add clarity to certain sections, "Metcut" and MetCAPP" may be used interchangeably.) Two major components of MetCAPP are used within GPPE. These are CUTTECH and CUTDATA. CUTTECH contains machining operation requirements, including tool selection, machining parameters and machining time computation formulas. These requirements are retrieved from MetCAPP by specifying one of a set of thirty-eight manufacturing features and its dimensions along with various material characteristics. CUTDATA operates at a lower level than CUTTECH, and provides access to a wealth of tool and machining information, such as feed and speed data.

For detailed information on the MetCAPP product refer to the "MetCAPP User's Guide."

3.3.6.1.2 Site Specific Process Knowledge

For those feature instances which cannot be handled by a simple one-to-one mapping from the manufacturing feature to a MetCAPP process, a more general mechanism is available. This mechanism allows any two or more MetCAPP operations to be combined and executed in sequence to create a single feature.

3.3.6.1.3 Process Knowledge Limitations and Assumptions

3.3.6.1.3.1 Reliance on MetCAPP

It is assumed at this time that all features must be handled by one or more MetCAPP processes. Unusual cases of features which require special processing will be handled using manufacturing rules. (See 3.3.6.1.3.2 "Representation of Non-MetCAPP Process Knowledge.")

MetCAPP was originally integrated into GPPE in order to sufficiently ground the "Micro" portion o the process plan in reality, assuring that the rough estimates required in the plan would be reliable. GPPE currently relies solely on the MetCAPP system for its Micro-process planning knowledge. Because of thi reliance, limitations in MetCAPP flexibility, tooling, material, and feature support translate to limitations i GPPE process model capabilities.

3.3.6.1.3.2 Representation of Non-MetCAPP Process Knowledge

In unusual cases of features which require special processing, or for simple features which ar simply not handled by MetCAPP, special processing using manufacturing rules is required. This proces knowledge can be represented in a variety of ways, but there are currently no industry standards to provic guidance. The minimum requirement for representing complicated, interacting rules for a problem such a tool selection or fixturing, will be a highly customized program. The capability to represent custom proce knowledge in the GPPE environment has not yet been exercised.

3.3.6.2 Process Model External Data

3.3.6.2.1 Machines

For each Resource Model machining workstation entry, there exists a similar MetCAPP entry that machine. The MetCAPP entry however, contains very detailed machine information including su items as machine limits, horsepower, feed rate capabilities, coolant options, and all other machi characteristics. This data is currently entered manually into the MetCAPP system using MetCAPP d entry facilities.

3.3.6.2.2 Tools

The MetCAPP database contains a detailed library of machining tools. This data may be ente and modified manually.

3.3.6.2.3 Materials

The MetCAPP database contains a sizable library of materials and their properties. Each mate... item in the Resource Model is mapped to a similar MetCAPP material.

3.3.6.2.3.1 MetCAPP Material Mapping

Materials which are called out in the Resource Model are mapped to the MetCAPP materi database by means of an ICAD catalog called "metcut-material- mapping.table." (See Appendix IV-C a sample metcut-material-mapping catalog.) As a new material is added to the "materials" catalog a made available in the Resource Model, an equivalent entry must be made in the "metcut-material- map catalog to assure that the appropriate material is used by MetCAPP in its computations. Each MetC material is specified by a family, an identifier, and a hardness. Therefore, the material mapping ca serves as a cross-reference which contains a MetCAPP material family-id-hardness reference for material in the Resource Model. It is not always the case that there is an exact match between a material in the Resource Model and a MetCAPP material. When this occurs, a material of equiv machinability (based on the MetCAPP family, identifier and hardness) is selected.

3.3.6.2.4 Process Knowledge

All other detailed process planning knowledge is contained in MetCAPP's internal rulebases.

3.3.6.3 Process Model Instantiation

The prime functions of the process model are to provide a feature-to-process mapping, a process-to-operation mapping, and to instantiate those dimensional and other values which are required to compute time standard information. This is accomplished in the following manner:

3.3.6.3.1 Feature to Process Mapping

3.3.6.3.1.1 Mapping to a MetCAPP or RAMP Process

For each generatively supported feature (see 3.3.3.3.2.1 "Features Supported in GPPE V1.5") there exists a process mapping defpart. For example, for a circular-face manufacturing feature, there exists a "circular-face-process- map" defpart. This process map contains children which point to one or more MetCAPP or RAMP processes by which this feature may be physically realized. For example, the circular-face feature lists "metcut-facing-through" and "metcut-cut- off" as possible processes to realize this feature. Table 3-3 contains a listing of generatively supported manufacturing features and their MetCAPP or RAMP process mapping equivalents in GPPE V1.5.

Table 3-3 Manufacturing Features and Their Process Mappings

Manufacturing Features	Metcut or RAMP Process
circular-face	metcut-facing-through
	metcut-cut-off
outer-diameter-with-no-restricted-sides	metcut-straight-turning-through
external-thread	metcut-external-threading
square-circular-od-groove	metcut-od-groove
	metcut-form-turning
blind-hole	metcut-single-diameter-hole
blind-inner-diameter	metcut-internal-holemaking
	metcut-single-diameter-hole
through-inner-diameter	metcut-internal-holemaking
	metcut-single-diameter-hole
through-hole	metcut-single-diameter-hole
circular-id-edge-flat	metcut-contour-turning
	ramp-groove-contour-turning
circular-od-edge-flat	metcut-contour-turning
	metcut-form-turning

3.3.6.3.1.2 Fulfilling Metcut Process Requirements

For each MetCAPP or RAMP process defined by a feature's process map, there exists a defpart which extracts all necessary feature data from the Manufacturing Part Model, and fulfills all MetCAPP protocol requirements. This entails building a list which contains all necessary dimensional and other data, along with the appropriate MetCAPP process callout. This information is eventually used to make the actual call to MetCAPP.

3.3.6.3.1.3 GPPE Interaction with MetCAPP

GPPE is interfaced with the MetCAPP system through the MetCAPP Applications Program Interface (API).

3.3.6.3.1.3.1 The MetCAPP API

The MetCAPP API is a means of interfacing with the MetCAPP system via IPC (inter-process communications). In the UNIX BSD environment, this is accomplished through Berkeley sockets.

When the *server* MetCAPP API is executed, it creates a dedicated socket, which is made available across the network for use by GPPE.

When GPPE requests the use of the MetCAPP API, it does so by referencing the dedicated socket by its handle. It is through this handle that all communications between the client and the server are accomplished (see Figure 3-20).

For further information regarding MetCAPP, and the MetCAPP API, refer to the "MetCAPP API Toolkit Programmer's Reference Manual."

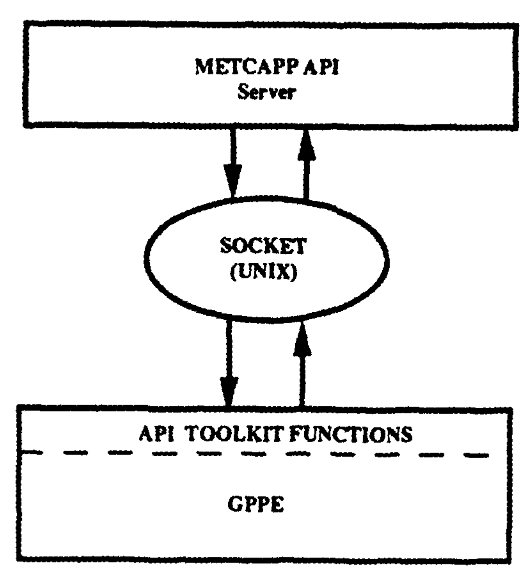


Figure 3-20 MetCAPP API Inter-process Communication Link

3.3.6.3.1.3.2 The GPPE MetCAPP API Interface Toolkit

The API Interface Toolkit is a component of MetCAPP consisting of a library of 'C' functions which provide the basic MetCAPP interaction mechanisms. GPPE implements its own version of the Interface Toolkit in which virtually all Toolkit functions are duplicated in Lisp. All calls to MetCAPP are realized through the execution of these Lisp functions. MetCAPP values are typically obtained in the following manner:

1) Set all required Metcut Variables: Key values such as material, feature and machine name are specified to Metcut through a series of "put" commands.
2) Order Metcut to perform an operation: Using the values which were specified, order Metcut to perform an action (such as computing operations and their details, selecting a tool, or searching for a material) via a "search" or other appropriate command.
3) Retrieve Results: Retrieve desired values via any of various "get" commands and assign the results to the appropriate attributes.

For full documentation of the Toolkit functions refer to the "MetCAPP API Toolkit Programmer's Reference Manual."

3.3.6.3.2 RAMP Processes

The RAMP Processes branch of the Process Model product structure tree is instantiated with every process which is defined by a sequence of MetCAPP processes (see Figure 3-21). The tree which is displayed includes a listing of each process and its associated machining classification.

The RAMP Processes branch is simply a special view of the processes available in the features branch of the Process Model product structure tree, and is used for inspection purposes only.

The RAMP Process and Metcut Processes branch of the Process Model, taken together, represent all processes which are supported within GPPE.

```
Process-Model
  Features
    ...
  Ramp-Processes
    Groove Contour Turning
      Turning
    Tapered Inner Diameter Turning
      Turning
  Metcut-Processes
    ...
```

Figure 3-21 RAMP Processes

3.3.6.3.3 Metcut Processes

The Metcut Processes branch of the Process Model product structure tree is instantiated with every process which is defined by a single MetCAPP process (see Figure 3-22). The tree which is displayed includes a listing of each process and its associated machining classification.

Like the RAMP Processes branch, the Metcut Processes branch is a view of the processes available in the features branch of the Process Model product structure tree and as such is used for inspection purposes only.

The RAMP Process and Metcut Processes branch of the Process Model, taken together, represent all processes which are supported within GPPE.

```
Process-Model
    Features
        ...
    Ramp-Processes
        ...
    Metcut-Processes
        Contour Turning
            Turning
        Cut Off
            Turning
        Facing To Center
            Turning
        Form Turning
            Turning
        Inner Diameter Turning (Drill, Bore, Etc.)
            Turning
        Holemaking (Drill, Bore, Ream, Etc.)
            Holemaking
        Outer Diameter Turning Through
            Turning
        Outer Diameter Turning To A Shoulder
            Turning
        Outer Diameter Grooving
            Turning
        Inner Diameter Threading Through
            Turning
        Inner Diameter Threading Blind
            Turning
        Tapped Hole - Blind
            Holemaking
        Tapped Hole - Through
            Holemaking
        Outer Diameter Threading
            Turning
```

Figure 3-22 Metcut Processes

3.3.6.4 Process Model Outputs

The Process Model itself exists as a data source for other modules, and as such, responds to requests for its own attribute data. (See 3.2.1.2.3 "The Demand-driven Nature of IDL.") Certain attribute requests will in turn execute MetCAPP which returns values for those attributes.

3.3.6.5 Process Model Limitations

3.3.6.5.1 Feature-to-Process Mapping Limitations

A. Tolerances

Feature tolerances are not yet considered automatically in the feature-to-process mapping functions. Tight tolerances should dictate different required operations. Appropriate consideration of the ability of a process to achieve the desired tolerances is currently undertaken by the user.

B. Machine Capabilities and Limitations

There is currently no automatic comparison of feature requirements to machine capabilities and limitations to determine the appropriateness of any given process mapping. As a result, a feature may be mapped to processes which are incapable of adequately realizing the feature, given all of its constraints. Some constraints might include machine rigidity, tolerance constraints and accessibility.

There is no automatic consideration of a part's size in the mapping from features to processes. An axisymmetric part may be too large to fit on any in-house turning center.

Appropriate consideration of these machine capabilities and limitations constraints is currently undertaken manually by the user.

C. Determining Tool Resource Availability and Suitability

Because tool capabilities have not yet been implemented in the Resource Model, there are currently no provisions to check that the ideal tool returned by MetCAPP is available at the given site. (See 3.3.5 "Resource Model Summary.")

3.3.6.5.2 MetCAPP Limitations

MetCAPP provides an excellent source of low level operation data and is very well suited to provide a basis for customized process formulas and logic. Currently however, high level MetCAPP functions are utilized extensively by the GPPE Process Model; thus, limitations in MetCAPP high-level knowledge translate to limitations in the Process Model. A few of the more critical limitations of MetCAPP follow:

1) MetCAPP does not include tolerance data in its selection of operations, tools, feeds, and speeds. Regardless of the tolerance of any given feature, MetCAPP provides the same operation sequence for the feature based simply on its geometric dimensions.

2) MetCAPP provides very little tool selection flexibility. MetCAPP provides cutting data on a feature by feature basis, and is not capable of considering a group of features concurrently. Because of this rigidity, MetCAPP often recommends

different "ideal" tools for each feature. Resulting in an unrealistic process plan for low volume orders.

MetCAPP cannot be directed to use a particular tool on a feature, but is only capable of selecting an "ideal" tool for the feature.

3) Each MetCAPP feature falls into one of three machining categories: milling, turning, or holemaking. There is currently no support for other types of machining operations, such as grinding, broaching, and others.

4) MetCAPP currently supports a set of 38 machining features. There are several common features which are not supported. These include: circular face-grooves, knurls, non-circular holes and tapered inner diameters.

3.3.7 Process Sequencer Summary

The Process Sequencer provides a structured interactive environment in which the user may select features, processes, machines and others to appropriately sequence a final macro process plan. The Sequencer utilizes the user interface capabilities of the ICAD Browser extensively to present the user with options, and allow their selection. The features available for manufacture are those features as specified in the features branch of the Manufacturing Part Model. The Process Sequencer makes extensive use of the Process Model, with MetCAPP being accessed frequently on a demand-driven basis to provide quantitative information regarding operations and operation details.

The Process Sequencer product structure tree consists of three major components (see Figure 3-23).

1) Sequence setup: In sequence setup, any standard tool or material preparations which are generally performed prior to any machining may be specified.

2) Machining setup: The machining setup branch may consist of a number of machining setups. It is within this branch that the machining operations which are required to realize each manufacturing feature are selected and ordered. With each machining setup, there are several user-directed options which may be specified including: the machining workstation, fixturing, lead-in operations and follow-on operations. Until all features have been machined, each machining setup branch contains a child representing the next setup. In this way, a setup to setup sequence is built.

3) Sequence Completion: The sequence completion component contains any common sequence completion steps which would be inserted into the process plan after all machining setups have been completed.

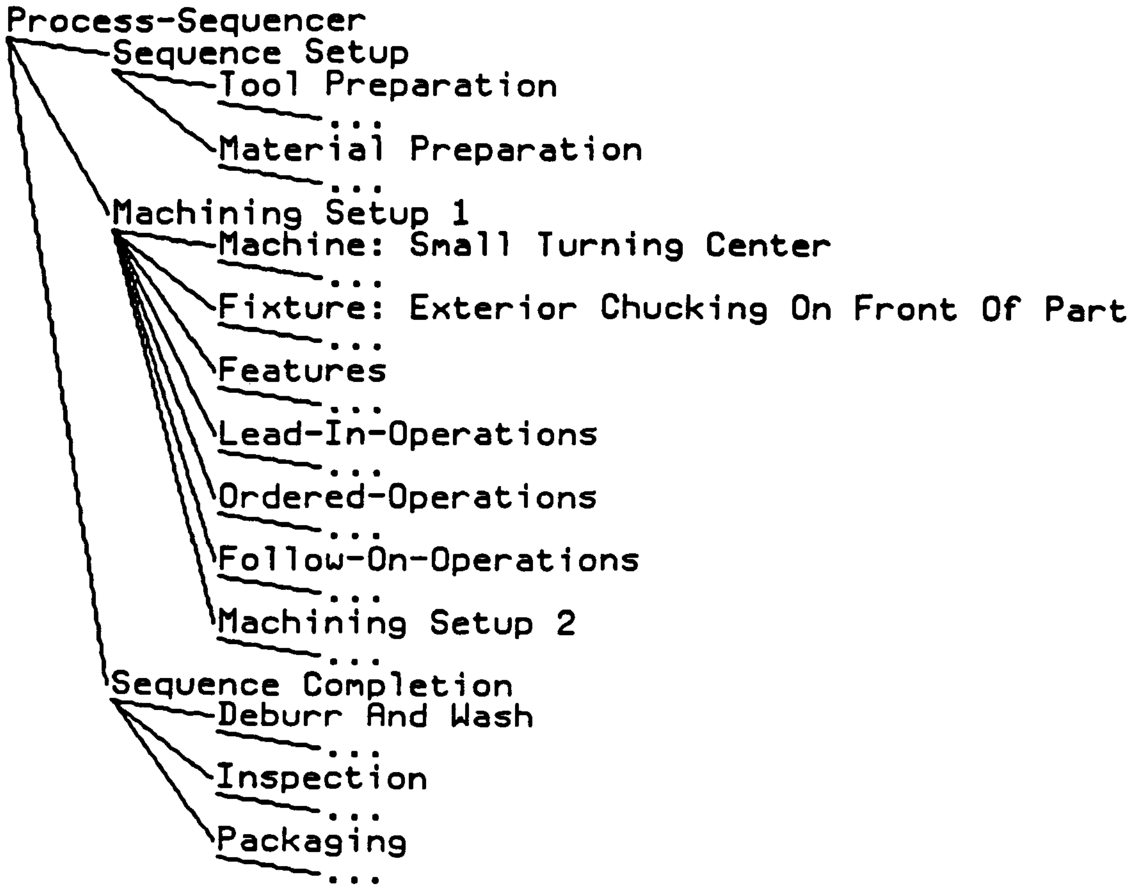


Figure 3-23 The Process Sequencer

3.3.7.1 Process Sequencer External Data

3.3.7.1.1 Time Standards

Times standards data used by the Process Sequencer is contained in the "Time-standards" catalog. (Refer to Appendix IV-D for a sample Time-standards catalog.) This catalog contains information regarding tool preparation, material preparation, fixture setup, lead-in operations, follow-on operations, deburr and wash operations, inspection operations, external operations, receiving, and packaging operations. The time-standards catalog is created through a series of SQL queries to the Manufacturing Engineering common database.

The time standards catalog is generally perused and internalized at each system start-up. It may however, be updated and compiled at any point. (See 3.2.1.4.1 "Catalogs" for more information regarding catalogs.)

3.3.7.2 Process Sequencer Instantiation

3.3.7.2.1 Sequence Setup

Sequence setup is instantiated with material preparation and tool preparation values as directed by the user. All available sequence setup options, along with their associated default time standards are defined in the "time-standards.table" catalog. (See 3.3.7.1.1 "Time Standards.")

Instantiating user-defined time-standard instructions which are wholly defined by the time-standards catalog is accomplished by simply querying the time-standards catalog for all options, presenting these to the user (see Figure 3-24), and allowing the user to select the desired options. The selected time standards then are taken directly from the catalog and placed in the product structure tree. In addition to the default available time standards, customized time standards may be defined by the user at any point. Sequence setup uses the following general method for instantiating data.

Begin

- Assign a value to the "workstation-information" attribute. This value is a list of the types of time standards which are to be ordered. For example, Material preparation or tool preparation.
- Append any required time standard values which must appear in the pop-up window to the "workstation-information" attribute.
- For each member of "workstation-information"

  Begin

  - Locate its "workstation-id" (a reference to the correct catalog data)
  - Query the time-standards table for all standards for the workstation-id
  - Present time standards to the user using choice-attributes
  - Collect all User choices

  End
- Prompt user for any user-defined time standards
- Present User with all selected time-standards for ordering
- Collect ordered list, and add it to the current time-standards structure

End

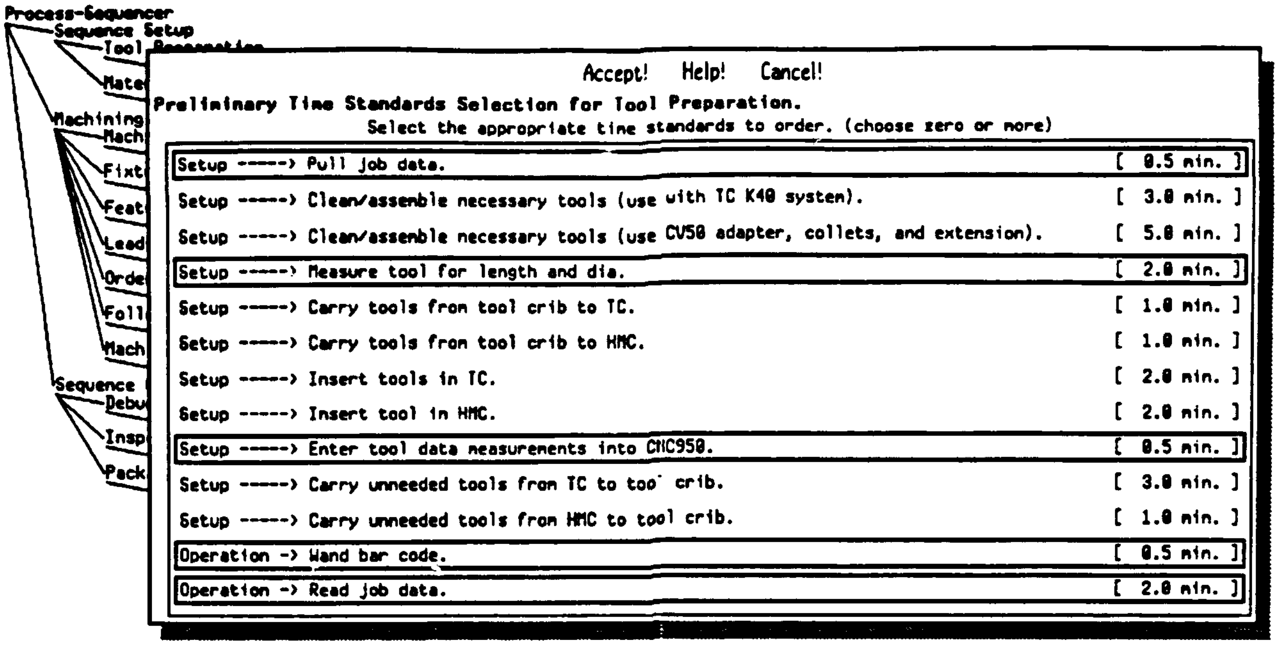


Figure 3-24 Time Standard Data is Extracted from the Catalog and Presented to the User for Selection and Ordering

3.3.7.2.2 Machining Setups

Any part may have from one to n machining setups where n = the number of manufacturing features in the part.

Defining the values which populate any given machining setup is largely an interactive process. The user selects the features to be machined in each setup, the process for each feature, the machine for the setup, the fixturing concept, and any lead-in or follow-on operations for each setup. MetCAPP is called to compute all operations and associated data for each process. After all operations have been computed, the user specifies the order in which they are to be executed.

The Machining Setup branch consists of six components (see Figure 3-25). Each component is described in the following sections.

3.3.7.2.2.1 Machine

The machine component of each machining setup is instantiated by allowing the user to select from a list of machines capable of realizing the features in the setup.

3.3.7.2.2.1.1 Workstation Selection

The workstation which will be utilized in a given setup is selected by the user from a set of recommended workstations. Selecting a valid set of recommended workstations is accomplished in two steps:

```
Machining Setup 1
  Machine: Small Turning Center
  Fixture: Exterior Chucking On Front Of Part
  Features
    (1) Circular Face At (0.0 5.125 0.0)
      Facing To Center
        Operations:
          1) Semi-Fin Face -- 2 Passes
            Trn-754-03
          2) Finish Face -- 1 Pass
            Trn-754-03
    (2) Outer Diameter With No Restricted Sides At (0.0 2.562 0.0)
      Outer Diameter Turning Through
        Operations:
          1) Normal Rough Turn -- 2 Passes
            Trn-754-03
          2) Finish Turn -- 1 Pass
            Trn-754-01
    (3) Circular Od Edge Flat At (0.0 5.109 0.0)
      Contour Turning
        Operations:
          1) Finish Contour -- 2 Passes
            Trn-758-03
    (4) Square Circular Od Groove At (0.0 2.57 0.0)
      Outer Diameter Grooving
        Operations:
          1) Finish Groove -- 1 Pass
            Grooving Tool
  Lead-In-Operations
    Trc Fixturing
      Setup -----> Read Job Data & Graphics.                          [ 3.5 Min. ]
        Setup -----> Placing By Hand To 3-Jaw Chuck.                    [ 0.4 Min. ]
  Ordered-Operations
    (1 Of 2) Semi-Fin Face The Circular Face At (0.0 5.125 0.0) -- 2 Passes
      (2 Of 2) Finish Face The Circular Face At (0.0 5.125 0.0) -- 1 Pass
        (1 Of 2) Normal Rough Turn The Outer Diameter With No Restricted Sides At (0.0 2.562 0.0) -- 2 Passes
          (2 Of 2) Finish Turn The Outer Diameter With No Restricted Sides At (0.0 2.562 0.0) -- 1 Pass
            (1 Of 1) Finish Contour The Circular Od Edge Flat At (0.0 5.109 0.0) -- 2 Passes
              (1 Of 1) Finish Groove The Square Circular Od Groove At (0.0 2.57 0.0) -- 1 Pass
  Follow-On-Operations
    Inspection
      Operation -> Use Visual Check To Inspect.                       [ 1.0 Min. ]
        Operation -> Input Dimensions To Workstation After Inspection.  [ 1.0 Min. ]
  Machining Setup 2
    ...
```

Figure 3-25 The Machining Setup

1) Compute Machine-class: For each feature in a given setup, the user selects a specific process to manufacture that feature. (See 3.3.7.2.2.3.1 "Selection of Features to be Manufactured in a Setup.") For every selected process there exists an *associated* ***machine-class*** *(either milling, turning, or holemaking)* which is capable of performing the process.

If all feature processes do not belong to the same machine class, then this indicates that the feature set cannot be manufactured on the same machine, and the user is required to select a different feature/process combination for the setup.

2) Determine Capable Machines: Once the machine-class has been determined, all machines in the Resource Model (see 3.3.5.2.1 "Workstations") which fit this machine class description are identified and are presented to the user for selection.

3.3.7.2.2.1.2 Workstation Selection Limitations

Valid machines are selected on the basis of machine class. Other parameters, such as horsepower limitations and travel limits are not automatically considered at this time. This limitation may cause extraneous machines which are not capable of realizing the features in the setup to be included in the list of valid machines. Proper consideration of machine capabilities and limitations is currently undertaken manually by the user.

3.3.7.2.2.2 Fixture

The fixture component of each machining setup is instantiated by allowing the user to select from a list of fixturing concepts which are valid for the current setup.

3.3.7.2.2.2.1 Fixture Selection

The set of valid fixturing concept options is identified in two steps:

1) Compute Machine-class: For each feature in a given setup, the user selects a specific process to manufacture that feature. (See 3.3.7.2.2.3.1 "Selection of Features to be Manufactured in a Setup.") For every process there exists an associated **machine-class** (either milling, turning, or holemaking) which is capable of performing the process.

   If all feature processes do not belong to the same machine class, then this indicates that the feature set cannot be manufactured on the same machine (or the with the same fixture), and the user is required to select a different feature/process combination for the setup.

2) Determine Appropriate Fixturing Concepts: Once the machine-class has been determined, all fixturing concepts in the Resource Model (see 3.3.5.2.3 "Fixtures") which fit this machine class description are identified and are presented to the user for selection.

3.3.7.2.2.2.2 Fixturing Limitations

Selection of fixturing for a setup is limited to the selection of a fixturing concept rather than an actual fixture. The fixture's orientation with respect to the part is specified in general terms by the user (for example, "Exterior chucking on rear of part"). Thus, the fixture's size, orientation and location are not

explicitly modeled or automatically considered in the planning process. Proper consideration of fixturing constraints is performed manually by the user.

Valid fixtures are selected on the basis of machine class. Other parameters, such as machine limitations and tolerances are not automatically considered at this time. This limitation may cause extraneous fixtures which are inappropriate, given the constraints, to be included in the list cf valid fixtures. Proper consideration of these constraints is currently undertaken manually by the user.

Fixture build-up sequences are currently not included in the final process plan.

3.3.7.2.2.3 Features

The features component of each machining setup is instantiated with the features to be manufactured together in a single setup. For each feature, the processes and operations required to realize each feature, the tool required for each operation, tool details, pass details (including the number of passes, time per pass, feeds and speeds) and all other information necessary to realize the feature at the machine are computed.

3.3.7.2.2.3.1 Selection of Features to be Manufactured in a Setup

The user selects one or more features for each setup. The group of valid manufacturing features available to each setup consists of a re-instantiation of all features found in the Manufacturing Part Model minus those features which have already been included in previous setups.

The valid feature set is presented to the user using choice attributes.

3.3.7.2.2.3.2 Selecting Processes

For each feature in a setup, a list of valid processes by which the feature may be manufactured is presented to the user for selection. (See 3.3.1.4 "Specifying Input Values.") This list of valid processes is compiled by performing a feature to process mapping for each feature.

3.3.7.2.2.3.2.1 Feature to Process Mapping

For each feature selected by the user to manufacture in the current setup, there exists a **process-model-feature** attribute. **Process-model-feature** refers to the feature's process map located in the Process Model. Refer to 3.3.6.3.1 "Feature to Process Mapping" for more detailed information on feature-to-process mapping. This reference to the Process Model's process map provides all process details required by the Process Sequencer.

It should be noted that this same feature-to-process mapping occurs within the Manufacturing Part Model. However, the attribute "suppress-expansion" causes the features and their process mappings to be displayed differently in the product structure tree. (See 3.3.4.2.1.1.2 "Manufacturing Part Model, Process and Machine Mapping.")

3.3.7.2.2.3.3 Operations

For each process selected by the user, the operations required to complete the process, along with the tool, pass detail, tool detail, tool-class, feed, speed, number of passes, time per pass, and total time

for each operation is computed. When the operation branch is expanded this is accomplished in the following manner:

```
Begin
 IF the process is a Metcut process
   begin
        Set all MetCAPP parameters
              This includes the following:
                   o    A MetCAPP material is selected based on the metcut-material-mapping
                        (see 3.3.6.2.3.1 "Metcut Material Mapping")
                   o    All required dimensions are set. This includes all geometric dimensions
                        and constraints necessary for the given MetCAPP process (see 3.3.6.3.1.2
                        "Fulfilling MetCAPP Process Requirements")
                   o    A valid MetCAPP machine is selected from the list of valid MetCAPP
                        machine tools. (This list is contained in the global variable
                        *metcut-machines*.)
   end
 ELSE (the process is a RAMP process)
        for each MetCAPP process
              begin
                   Set all Metcut parameters (as above)
              end
  endif
 Call MetCAPP with all of the appropriate parameters
 For each Operation returned by the MetCAPP call
        begin
              Instantiate operation
              Assign results to the appropriate attributes
        end
End
```

3.3.7.2.2.3.4 Operation Ordering

The operation ordering branch of each setup is instantiated with the ordered sequence of operations for the setup's feature set (see Figure 3-25). The operations are sequenced interactively by the user. This is accomplished in the following manner:

```
Begin
For each feature in the setup
        operation-object-list = list of all MetCAPP operations for the feature
While User has not indicated "Done"
  begin
        IF there are remaining non-nil operation-object-lists
              remaining-operations = the first element of
                   each operation-object-list + "non-machining operations"
        ELSE (all machining operations have been sequenced)
              remaining-operations = "non-machining operations" + "Done"
        Prompt user to select next operation from remaining-operations
        IF user selection is "non-machining operations"
              Prompt user with non-machining operations list (derived from Resource Model)
        ELSE (User selected a machining operation)
          begin
              Remove selected operation from remaining-operations
              IF operation-object-list for the feature is not empty
```

Remove the first member from the operation-object-list

end

Instantiate user's selection as a child of the previous operation

End

3.3.7.2.2.3.5 Lead-in Operations

The lead-in operations component of each machining setup is instantiated in exactly the same manner as was the sequence setup. (See 3.3.7.2.1 "Sequence setup.") However, in this component, the "workstation-information" attribute is instantiated through user selection. The lead-in operations options are derived from the Resource Model non-machining workstation information, which was originally derived from the workstation-info catalog. (See 3.3.5.2.1 "Resource Model, Workstations.")

3.3.7.2.2.3.6 Follow-on Operations

The follow-on operations component of each machining setup is instantiated in exactly the same manner as was the sequence setup. (See 3.3.7.2.1 "Sequence Setup.") However, in this component, the "workstation-information" attribute is populated through user selection. The valid follow-on operations options are derived from the Resource Model non-machining workstation information, which was originally derived from the workstation-info catalog. (See 3.3.5.2.1 "Resource Model, Workstations.")

3.3.7.2.3 Sequence Completion

The sequence completion component of the Process Sequencer is instantiated in exactly the same manner as was the Sequence setup. (See 3.3.7.2.1 "Sequence Setup.") The values assigned to the "workstation-information" attribute for sequence completion are: Deburr and Wash, Inspection, and Packaging.

The valid sequence completion options are derived from the Resource Model non-machining workstation information, which was originally derived from the workstation-info catalog. (See 3.3.5.2.1 "Resource Model, Workstations.")

3.3.7.3 Process Sequencer Outputs

Like the PIF (see 3.3.4.2.3.2 "PIF Generation") the Process Plan and Tool Details are generated through the use of companions and writers which traverse the Process Sequencer product structure tree, gather the appropriate attribute values, and format them accordingly.

3.3.7.3.1 Process Plan

The primary output of the Process Sequencer is the process plan itself. The process plan may be output in two formats:

1) Routing: The routing file is output in a tabular format. This format includes information about each setup, and the operations, along with its setup, move, and execution time. (For a sample process plan routing refer to Appendix V-B.)

2) Description: This man-readable output format includes detailed information regarding tools, passes, feeds, speeds, and others. (For a sample description refer to Appendix V-A.)

3.3.7.3.2 Tool Details

The process sequencer may output tool detail files for each tool referenced in the plan. (Refer to Appendix V-C for sample tool detail file output.)

3.3.7.3.3 Exceptions

The process sequencer may output an exception report file. The exception report is a free-form text file which may contain any user-entered data.

This page intentionally left blank.

APPENDIX I

A SAMPLE PDES FILE

```
DATA;
@220 = SCALED_LENGTH_UNIT(4, 1.000000);
@240 = SCALED_MASS_UNIT(3, ());
@260 = SCALED_TIME_UNIT(1, ());
@280 = SCALED_CURRENT_UNIT(0, ());
@300 = SCALED_TEMPERATURE_UNIT(1, ());
@320 = SCALED_AMOUNT_UNIT(0, ());
@340 = SCALED_LUMINOUS_INTENSITY_UNIT(0, ());
@360 = SCALED_PLANE_ANGLE_UNIT(2, ());
@380 = SCALED_SOLID_ANGLE_UNIT(0, ());
@400 = UNITS(#220, #240, #260, #280, #300, #320, #340, #360, #380);
@420 = DIRECTION((), 0.000000, 0.000000, 1.000000);
@440 = DIRECTION((), 1.000000, 0.000000, 0.000000);
@460 = CARTESIAN_POINT((), 0.000000, 0.000000, 0.000000);
@480 = AXIS2_PLACEMENT((), #460, #420, #440);
@500 = CIRCLE((), 0.997500, #480);
@520 = CARTESIAN_POINT((), 0.997500, 0.000000, 0.000000);
@540 = CARTESIAN_POINT((), -0.997500, 0.000000, 0.000000);
@560 = TRIMMED_CURVE((), #500, 0.000000, 3.141593, #520, #540, .T.);
@580 = CURVE_LOGICAL_STRUCTURE(#560, .T.);
@600 = VERTEX(#520);
@620 = VERTEX(#540);
@640 = EDGE(#600, #620, #580);
@660 = EDGE_LOGICAL_STRUCTURE(#640, .T.);
@820 = TRIMMED_CURVE((), #500, 0.000000, 3.141593, #540, #520, .T.);
@840 = CURVE_LOGICAL_STRUCTURE(#820, .T.);
@900 = EDGE(#620, #600, #840);
@920 = EDGE_LOGICAL_STRUCTURE(#900, .T.);
@940 = EDGE_LOOP((#660, #920));
@960 = LOOP_LOGICAL_STRUCTURE(#940, .T.);
@980 = FACE(#960, (#960), (), ());
@1000 = FACE_LOGICAL_STRUCTURE(#980, .F.);
@1060 = CARTESIAN_POINT((), 0.000000, 0.000000, 5.125000);
@1080 = AXIS2_PLACEMENT((), #1060, #420, #440);
@1100 = CIRCLE((), 0.997500, #1080);
@1120 = CARTESIAN_POINT((), 0.997500, 0.000000, 5.125000);
@1140 = CARTESIAN_POINT((), -0.997500, 0.000000, 5.125000);
@1160 = TRIMMED_CURVE((), #1100, 0.000000, 3.141593, #1120, #1140, .F.);
@1180 = CURVE_LOGICAL_STRUCTURE(#1160, .F.);
@1200 = VERTEX(#1120);
@1220 = VERTEX(#1140);
@1240 = EDGE(#1200, #1220, #1180);
@1260 = EDGE_LOGICAL_STRUCTURE(#1240, .F.);
@1420 = TRIMMED_CURVE((), #1100, 0.000000, 3.141593, #1140,  #1120, .F.);
@1440 = CURVE_LOGICAL_STRUCTURE(#1420, .F.);
@1500 = EDGE(#1220, #1200, #1440);
```

```
@1520 = EDGE_LOGICAL_STRUCTURE(#1500, .F.);
@1540 = EDGE_LOOP((#1260, #1520));
@1560 = LOOP_LOGICAL_STRUCTURE(#1540, .T.);
@1580 = FACE(#1560, (#1560), (), ());
@1600 = FACE_LOGICAL_STRUCTURE(#1580, .T.);
@1760 = TRIMMED_CURVE((), #500, 0.000000, 3.141593, #520, #540, .F.);
@1780 = CURVE_LOGICAL_STRUCTURE(#1760, .F.);
@1840 = EDGE(#600, #620, #1780);
@1860 = EDGE_LOGICAL_STRUCTURE(#1840, .F.);
@1920 = POLYLINE((), (#520, #1120));
@1940 = CURVE_LOGICAL_STRUCTURE(#1920, .T.);
@2000 = EDGE(#600, #1200, #1940);
@2020 = EDGE_LOGICAL_STRUCTURE(#2000, .T.);
@2180 = TRIMMED_CURVE((), #1100, 0.000000, 3.141593, #1120, #1140, .T.);
@2200 = CURVE_LOGICAL_STRUCTURE(#2180, .T.);
@2260 = EDGE(#1200, #1220, #2200);
@2280 = EDGE_LOGICAL_STRUCTURE(#2260, .T.);
@2340 = POLYLINE((), (#540, #1140));
@2360 = CURVE_LOGICAL_STRUCTURE(#2340, .F.);
@2420 = EDGE(#620, #1220, #2360);
@2440 = EDGE_LOGICAL_STRUCTURE(#2420, .F.);
@2460 = EDGE_LOOP((#1860, #2020, #2280, #2440));
@2480 = LOOP_LOGICAL_STRUCTURE(#2460, .T.);
@2500 = SUBFACE(#2480, (#2480), ());
@2660 = TRIMMED_CURVE((), #500, 0.000000, 3.141593, #540, #520, .F.);
@2680 = CURVE_LOGICAL_STRUCTURE(#2660, .F.);
@2740 = EDGE(#620, #600, #2680);
@2760 = EDGE_LOGICAL_STRUCTURE(#2740, .F.);
@2820 = POLYLINE((), (#540, #1140));
@2840 = CURVE_LOGICAL_STRUCTURE(#2820, .T.);
@2900 = EDGE(#620, #1220, #2840);
@2920 = EDGE_LOGICAL_STRUCTURE(#2900, .T.);
@3080 = TRIMMED_CURVE((), #1100, 0.000000, 3.141593, #1140, #1120, .T.);
@3100 = CURVE_LOGICAL_STRUCTURE(#3080, .T.);
@3160 = EDGE(#1220, #1200, #3100);
@3180 = EDGE_LOGICAL_STRUCTURE(#3160, .T.);
@3240 = POLYLINE((), (#520, #1120));
@3260 = CURVE_LOGICAL_STRUCTURE(#3240, .F.);
@3320 = EDGE(#600, #1200, #3260);
@3340 = EDGE_LOGICAL_STRUCTURE(#3320, .F.);
@3360 = EDGE_LOOP((#2760, #2920, #3180, #3340));
@3380 = LOOP_LOGICAL_STRUCTURE(#3360, .T.);
@3400 = SUBFACE(#3380, (#3380), ());
@3420 = EDGE_LOOP((#2020, #1860, #3180, #3340, #2760, #2280));
@3440 = LOOP_LOGICAL_STRUCTURE(#3420, .F.);
@3460 = FACE(#3440, (#2480, #3380, #3440), (), (#3400, #2500));
@3480 = FACE_LOGICAL_STRUCTURE(#3460, .T.);
@3500 = CLOSED_SHELL((#1600, #3480, #1000));
@3520 = SHELL_LOGICAL_STRUCTURE(#3500, .T.);
@3540 = MANIFOLD_SOLID_BREP(#3520, ());
@3560 = BREP_AREA_REP(#980, #3540);
@3580 = MACHINING_ALLOWED_SURFACE_TEXTURE((), (), 0.000250, 4, 0.030000, 4, (), (), (), (), (),
(), (), (), ());
@3600 = MAXIMAL_AREA_SHAPE_ELEMENT((), (#3580), (#3560), (), ());
```

@3620 = BREP_AREA_REP(#1580, #3540);
@3660 = MAXIMAL_AREA_SHAPE_ELEMENT((), (#3580), (#3620), (), ());
@3680 = BREP_NM_AREA_REP(#2500, #3540);
@3700 = MACHINING_ALLOWED_SURFACE_TEXTURE((), (), 0.000063, 4, 0.030000, 4, (), (), (), (), (), (), (), (), ());
@3720 = NONMAXIMAL_AREA_SHAPE_ELEMENT((), (#3700), (), (), (#3680));
@3740 = BREP_NM_AREA_REP(#3400, #3540);
@3780 = NONMAXIMAL_AREA_SHAPE_ELEMENT((), (#3700), (), (), (#3740));
@3800 = BREP_AREA_REP(#3460, #3540);
@3820 = MAXIMAL_AREA_SHAPE_ELEMENT((), (#3700), (#3800), (), ());
@3840 = COORDINATE_TOLERANCE_RANGE(0.015625, 0.015625);
@3860 = LOCATION_DIMENSION(#3840, #3600, #3660, .F., .T., ());
@3880 = COORDINATE_TOLERANCE_RANGE(0.000000, 0.001000);
@3900 = SIZE_FEATURE((#3820));
@3960 = LINE((), #460, #420);
@3980 = SIZE_CHARACTERISTIC_DIMENSION(#3880, 0, #3900, #3960);
@4020 = DIRECTION((), 0.000000, 1.000000, 0.000000);
@4060 = AXIS2_PLACEMENT((), #1060, #4020, #420);
@4080 = GEOMETRIC_DERIVATION(.U., (), (#4060), ());
@4100 = UNCONDITIONED_DATUM(#4080, 'DTM1');
@4140 = DIRECTION((), -1.000000, 0.000000, 0.000000);
@4180 = AXIS2_PLACEMENT((), #1060, #4140, #420);
@4200 = GEOMETRIC_DERIVATION(.U., (), (#4180), ());
@4220 = UNCONDITIONED_DATUM(#4200, 'DTM2');
@4240 = CARTESIAN_POINT((), 0.000000, 0.000000, 2.570312);
@4260 = DIRECTION((), 0.000000, 0.000000, -1.000000);
@4300 = AXIS2_PLACEMENT((), #4240, #4260, #440);
@4320 = COORDINATE_TOLERANCE_RANGE(0.007812, 0.000000);
@4340 = SIZE_PARAMETER(#4320, 0, 0.359375);
@4360 = COORDINATE_TOLERANCE_RANGE(0.500000, 0.500000);
@4380 = ANGLE_PARAMETER(#4360, 0, 90.000000);
@4429 = COORDINATE_TOLERANCE_RANGE(0.010000, 0.010000);
@4430 = SIZE_PARAMETER(#4429, 1, 0.093000);
@4431 = IMPLICIT_EDGE_ROUND((), #4430);
@4440 = SQUARE_U_FEATURE_SWEEP_PROFILE(#4340, #4431, #4431, #4380, #4380);
@4460 = COORDINATE_TOLERANCE_RANGE(0.010000, 0.010000);
@4480 = SIZE_PARAMETER(#4460, 0, 1.625000);
@4500 = ANGLE_PARAMETER((), 0, 90.000000);
@4520 = COMPLETE_CIRCULAR_FEATURE_SWEEP_PATH(#4480, #4500);
@4540 = ALONG_FEATURE_SWEEP(#4300, #4520, #4440, ());
@4550 = IMPLICIT_FEATURE_BOUND((#3820));
@4560 = IMPLICIT_DEPRESSION(#4540, #4550, (), ());
@4580 = FORM_FEATURE('Imp Depr, Sq U Profile, Circular path, normal: SQU_N1', (#4560), (), (), ());
@4620 = NONMAXIMAL_AREA_SHAPE_ELEMENT(#4580, (#3700), (), (), ());
@4640 = GEOMETRIC_DERIVATION(.U., (), (#4300), ());
@4680 = LOCATION_DIMENSION(#4460, #4580, #4640, .T., .T., ());
@4700 = COORDINATE_TOLERANCE_RANGE(0.015625, 0.000000);
@4720 = LOCATION_DIMENSION(#4700, #4580, #3660, .T., .T., ());
@4740 = EDGE_BLENDED_INTERSECTION(#3660, #3820);
@4760 = EDGE_BLENDED_INTERSECTION(#3600, #3820);
@4780 = COORDINATE_TOLERANCE_RANGE(0.500000, 0.500000);
@4800 = ANGLE_PARAMETER(#4780, 0, 45.000000);
@4820 = COORDINATE_TOLERANCE_RANGE(0.015625, 0.015625);
@4840 = SIZE_PARAMETER(#4820, 0, 0.031250);

```
@4860 = IMPLICIT_EDGE_FLAT((#4760, #4740), #4800, #4840);
@4880 = FORM_FEATURE('Edge Flat', (#4860), (), (), ());
@4900 = NONMAXIMAL_AREA_SHAPE_ELEMENT(#4880, (), (), (), ());
@4920 = IMPLICIT_MARKING(#3600, "423498-1' and contract number');
@4940 = MARKING((), (), #4920, (), (), ());
@4960 = FORM_FEATURE('Marking', (#4920), (), (), ());
@4980 = NONMAXIMAL_AREA_SHAPE_ELEMENT(#4960, (#4940), (), (#3600), ());
@5000 = IMPLICIT_MARKING(#3660, 'NGL');
@5020 = MARKING((), (), #5000, (), (), ());
@5040 = FORM_FEATURE('Marking', (#5000), (), (), ());
@5060 = NONMAXIMAL_AREA_SHAPE_ELEMENT(#5040, (#5020), (), (#3660), ());
@6060 = SHAPE((#3600, #3660, #3720, #3780, #3820, #4620, #4900, #4980, #5060));
@6080 = PART_MODEL(#3540, #400, #6060, (#4580, #4880, #4960, #5040), (), (#3860, #3980, #4680,
#4720), ());
@6100 = SPEC_CLASS((), ());
@6120 = GENERAL_NOTE(#6100, (), 'Break all sharp edges', (), (), ());
@6140 = SPEC_CLASS(0, 'QQ-P-416');
@6160 = MATERIAL_COMPOSITION('CADMIUM', 47, (), (), ());
@6180 = CHEMICAL_PROPERTY(#6140, 'CLASS 2, TYPE II', 1, 'PLATE', (), (#6160), 0);
@6200 = SPEC_CLASS(0, 'MIL-H-6875');
@6220 = HEAT_TREAT(#6200, 'RHN C40 TO C44', (), ());
@6240 = SPEC_CLASS(0, 'MIL-I-6868');
@6260 = INSPECTION_PROPERTY(#6240, '100% MAGNETIC PARTICLE', 'SHALL BE PERFORMED
BY BOTH CIR. AND LONG. METHOD. THE WET OR DRY PROCESS', 'WITH EITHER FLUOR OR
NON-FLUOR. PARTICLES SHALL BE USED.', 'NO CRACKS WILL BE ALLOWED.', ());
@6300 = GENERAL_NOTE(#6100, (), 'STOCK IS NOMINAL SIZE WITHOUT MANUFACTURING
ALLOWANCE, FOR NAEC USE ONLY.', (), (), ());
@6320 = SPEC_CLASS(0, 'MIL-S-5000, COND C2 OR C4');
@6340 = MATERIAL_PROPERTY((#6320), (), ());
@6360 = DATE(91, 6, 5);
@6380 = TIME(14, 19, 44, ());
@6400 = DATE_TIME(#6360, #6380);
@6420 = PERSON_NAME('E. Browne', 'Supervisor');
@6440 = PERSON_AND_ORGANIZATION(#6420, 'U.S.Navy', 'Engineering Department (SI)', 'Naval Air
Engineering Center', 'Lakehurst, NJ  00733');
@6460 = PRODUCT_ITEM_VERSION_FUNCTIONAL_DEFINITION(#6080, (#6340), (#6120, #6180, #6220,
#6260, #6300), (), #6440, 'RPGS', 'RPGS Development', #6400);
@6480 = PRODUCT_ITEM_VERSION('Grooved Pin', 'Flight Deck Hardware 3 1/2 Launching Shuttle',
#6440, #6400, (), (), (), (), 'NR', (), (#6460));
@6500 = PRODUCT_ITEM('GROOVED PIN', 'Grooved Pin', '80020_423498-1', (#6480));
ENDSEC;
```

APPENDIX II

PDES REPRESENTATION GENERATED BY CREATE-ICAD-FILE

```
(221 SCALED-LENGTH-UNIT (:unit (constant 4)
                         :scale-factor (real 1.000000) ) )
(241 SCALED-MASS-UNIT (:unit (constant 3)
                       :scale-factor () ) )
(261 SCALED-TIME-UNIT (:unit (constant 1)
                       :scale-factor () ) )
(281 SCALED-CURRENT-UNIT (:unit (constant 0)
                          :scale-factor () ) )
(301 SCALED-TEMPERATURE-UNIT (:unit (constant 1)
                              :scale-factor () ) )
(321 SCALED-AMOUNT-UNIT (:unit (constant 0)
                         :scale-factor () ) )
(341 SCALED-LUMINOUS-INTENSITY-UNIT (:unit (constant 0)
                                     :scale-factor () ) )
(361 SCALED-PLANE-ANGLE-UNIT (:unit (constant 2)
                              :scale-factor () ) )
(381 SCALED-SOLID-ANGLE-UNIT (:unit (constant 0) :scale-factor () ) )
(581 CURVE-LOGICAL-STRUCTURE (:curve-element (trimmed-curve 561)
                              :flag (logical t) ) )
(601 VERTEX (:vertex-point (cartesian-point 521) ) )
(621 VERTEX (:vertex-point (cartesian-point 541) ) )
.
.
(4441 SQUARE-U-FEATURE-SWEEP-PROFILE
    (:sweep-width (size-parameter 4341)
     :blend1 (implicit-edge-round 4432)
     :blend2 (implicit-edge-round 4432)
     :angle1 (angle-parameter 4381)
     :angle2 (angle-parameter 4381) ) )
(4481 SIZE-PARAMETER (:range
                      (coordinate-tolerance-range 4430) :measurement
                      (integer 0) :nominal-size
                      (real 1.625000) ) )
(4501 ANGLE-PARAMETER (:range ()
                       :measurement (integer 0)
                       :nominal-angle (real 90.000000) ) )
(4521 COMPLETE-CIRCULAR-FEATURE-SWEEP-PATH
    (:path-size (size-parameter 4481)
     :orientation-angle (angle-parameter 4501) ) )
(4541 ALONG-FEATURE-SWEEP
    (:location (axis2-placement 4301)
     :sweep-path (complete-circular-feature-sweep-path 4521)
     :sweep-profile (square-u-feature-sweep-profile 4441)
```

```
    :sweep-ends () ) )
(4551 IMPLICIT-FEATURE-BOUND (:elements ((maximal-area-shape-element 3821))))
(4561 IMPLICIT-DEPRESSION
    (:definition
       (along-feature-sweep 4541)
       :end-bound (implicit-feature-bound 4551)
       :end-blend ()
       :interruptions () ) )
.
.
(6341 MATERIAL-PROPERTY (:specification-type
                        ((spec-class 6321) )
                        :material-type ()
                        :material-condition () ) )
(6361 DATE (:year (integer 91)
         :month (integer 6)
         :day (integer 5) ) )
(6381 TIME (:hour (integer 14)
         :minute (integer 19)
         :second (integer 44)
         :msec () ) )
(6401 DATE-TIME (:dt (date 6361) :tm (time 6381) ) )
(6421 PERSON-NAME (:name-text (string "E. Browne")
                :title (string "Supervisor") ) )
(6441 PERSON-AND-ORGANIZATION (
                           :person (person-name 6421)
                           :company (string "U.S.Navy")
                           :department (string "Engineering Department (SI)")
                           :section (string "Naval Air Engineering Center")
                           :project (string "Lakehurst, NJ  00733") ) )
(6461 PRODUCT-ITEM-VERSION-
    FUNCTIONAL-DEFINITION (
                         :specific-functionality (part-model 6081)
                         :materials ((material-property 6341) )
                         :notes-and-specs
                         ((general-note 6121)
                          (chemical-property 6181)
                          (heat-treat 6221)
                          (inspection-property 6261)
                          (general-note 6301) )
                         :higher-assembly ()
                         :owner (person-and-organization 6441)
                         :name (string "RPGS")
                         :description (string "RPGS Development")
                         :creation-date (date-time 6401) ) )
(6481 PRODUCT-ITEM-VERSION (
                         :name (string "Grooved Pin")
                         :description
                         (string "Flight Deck Hardware 3 1/2 Launching Shuttle")
                         :creator (person-and-organization 6441)
                         :creation-date (date-time 6401)
                         :security-class ()
                         :approvals ()
                         :prior-version ()
```

```
                    :change-reason ()
                    :version-id (string "NR")
                    :contracts ()
                    :definitions
                    ((product-item-version-
                      functional-definition 6461) ) ) )
(6501 PRODUCT-ITEM (:name (string "GROOVED PIN")
                :description (string "Grooved Pin")
                :item-id (string "80020_423498-1")
                :versions ((product-item-version 6481) ) ) )
```

This page intentionally left blank.

APPENDIX III

SAMPLE PIF

```
CIRC
890
CIRC
590
CYL
1830 890 590 1.9950 0.0000 0.0010 1 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000 5.1250 0.0156 0.0156
SCOG
2390 1.6250 0.0100 0.0100 1 0.1850 0.0000 0.0000 0.3594 0.0156 0.0000 90.0000 0.5000 0.5000 90.0000
0.5000 0.5000 0.0930 0.0100 0.0100 0.0930 0.0100 0.0100 0.0000 2.5703 0.0000 0.0000 -1.0000 0.0000
CIRCHAM
2470 1.9325 1 45.0000 0.5000 0.5000 0.0313 0.0156 0.0156 0.0000 5.1250 0.0000 0.0000 -1.0000 0.0000
CIRCHAM
2480 1.9325 1 45.0000 0.5000 0.5000 0.0313 0.0156 0.0156 0.0000 0.0000 0.0000 0.0000 1.0000 0.0000
PDATUM
2250 1 0.0000 2.5703 0.0000 0.0000 -1.0000 0.0000 0.0000 0.0000 -1.0000
LDIM
2250 2390 0.0100 0.0100
LDIM
890 2390 0.0156 0.0156
ENOTE
890 Marking:  NGL
ENOTE
590 Marking:  '423498-1' and contract number
SURFIN
2390 63
SURFIN
1830 63
SURFIN
890 250
SURFIN
590 250
PNOTE
PLATE (PLATE) per MIL-SPEC QQ-P-416, before evaluating dimensions (CLASS 2, TYPE II)
PNOTE
INSPECT IN ACCORDANCE WITH Mil-I-6868. 100% MAGNETIC PARTICLE SHALL BE PERFORMED
BY BOTH CIR. AND LONG. METHOD.  THE WET OR DRY PROCESS WITH EITHER FLUOR OR
NON-FLUOR. PARTICLES SHALL BE USED. NO CRACKS WILL BE ALLOWED.
PNOTE
STOCK IS NOMINAL SIZE WITHOUT MANUFACTURING ALLOWANCE, FOR NAEC USE ONLY.
PNOTE
Break sharp edges
 PNOTE
Material:  PER  MIL-SPEC MIL-S-5000, COND C2 OR C4;
TBLOCK
80020 423498-1  GROOVED PIN
```

This page intentionally left blank.

APPENDIX IV

A. SAMPLE WORKSTATION CATALOG

```
catalog
{(:set-package "RAMP")}
{(:defcatalog *-WORKSTATION-INFO-*)}
{(:defline
  ( :workstation-id              ;symbol
    :base-operation-code                ;integer
    :workstation-desc            ;string
    :workstation-group                  ; One of :NON-MACHINING :MACHINING
:EXTERNAL-PROCESS
    :workstation-class           ; One of :MILLING, :TURNING, or :N/A
    :metcut-machine-group        ; e.g. turning, milling. Not applicable to external or non-machining.
    :metcut-machine-name         ; id used (at least for) connecting with METCAPP
  ))}
{(:prepare-to-read)}

tlbs01  110   "Tool Preparation"          :NON-MACHINING    :N/A          "Mechanical
Assembly" "Mechanical Assembly"
mtps01      210    "Material Preparation"      :NON-MACHINING    :N/A         "Sawing"
     "Sawing"
fixs01  310   "HMC Fixturing"             :NON-MACHINING    :N/A          "Milling"
"MILACRON T-10"
fixs02  360   "TRC Fixturing"             :NON-MACHINING    :N/A          "Turning"
"CINTURN 8C"
hmcs01      410    "Small Horizontal Machining Center"   :MACHINING   :MILLING    "Milling"
"MILACRON T-10"
hmcs01      410    "Small Horizontal Holemaking Center"  :MACHINING   :HOLEMAKING
"Holemaking"  "MILACRON T-10"
;DRR    340    "Rockwell - Drill/Tap"        :MACHINING                  :HOLEMAKING
"Holemaking"  "ROCKWELL MODEL 15"
;DSR    340 "Acme Drill-Single Spindle"       :MACHINING                 :HOLEMAKING
"Holemaking"  "ACME SS DRILL"
tncs01  510   "Small Turning Center"      :MACHINING        :TURNING      "Turning"
"CINTURN 8C"
dbrs01  610   "Deburr and Wash"           :NON-MACHINING    :N/A          "Hand Work"
"Hand Work"
inss01  810   "Inspection"                :NON-MACHINING    :N/A
"Inspection"  "Inspection"
prps01  910   "Packaging"                 :NON-MACHINING    :N/A          "Miscellaneous
Protective" "Miscellaneous Protective"
recs01  1110  "Receiving"                 :NON-MACHINING    :N/A          "Material Handling"
"Material Handling"
exts01  1010  "External Processing"       :EXTERNAL-PROCESS       :N/A        nil
     nil
```

B. SAMPLE MATERIALS CATALOG

```
CATALOG
{(:set-package "RAMP")}
{(:defcatalog *-MATERIALS-*)}
{(:defline
 (:ramp-id :description :shape :material-type :diameter :width :thickness
        :units :hardness-scale :hardness :mil-std :aisi-number
        :astm-number :qq-spec :qq-class :qq-condition :qq-form
        :cold-drawn? :hot-rolled? :annealed? :stress-relieved?
        :age-hardened? :hot-finished? :temper :material-designator))}

{(:prepare-to-read)}

M00000000000001 "bar rd monel-k 4.500 dia"    round "monel-k"    4.500 0  0
inchs bhn "245" "" "" "" "qq-n-286" "a" "" "2" nil nil t nil t t "n/a"
   "kmh"
M00000000000002 "bar rd monel-h 5.000 dia"    round "monel-h"    5.000 0  0
inchs bhn "240-290" "" "" "" "qq-n-288" "copm b" "as cast" "" nil nil nil nil nil
nil "n/a"   "nch"
M00000000000003 "bar rd copper alloy 3.125 dia" round "high leaded tin bronze" 3.125 0  0
inchs n/a "n/a" "" "" "b584 932" "qq-c-390" "c93200" "e7" "" nil nil nil nil nil
nil "n/a"   "cugn"
M00000000000005 "bar rd copper alloy 8.500 dia" round "leaded/non-leaded tin bronze" 8.500 0
 0  inchs n/a "n/a" "" "" "b505" "qq-c-390" "c90500" "d6" "" nil nil nil
nil nil nil "n/a"   "cugg"
M00000000000007 "bar rd steel 5.250 dia"    round "steel"    5.250 0  0
inchs bhn "269-321" "" "4140" "a322" "" "" "" "" nil t nil t nil t
"269-321 bhn" "cstw"
M00000000000008 "bar rd monel-k 1.375 dia"    round "monel-k"    1.375 0  0  inchs
 bhn "265" "" "" "" "qq-n-2863" "a" "" "2" nil t nil nil t t "n/a"
   "kmj"
M00000000000010 "bar rd brass 4.250 dia"    round "brass"    4.250 0  0
inchs rb "82" "" "" "" "qq-b-637" "464" "" "bar" nil t nil nil nil t
"half_hard" "brdf"
M00000000000011 "bar rd alum 3.500 dia"    round "aluminum"    3.500 0  0  inchs
 n/a "n/a" "mil-f-17132" "" "b211" "qq-a-225/8" "6061" "" "" t nil nil nil nil nil "t6"
   "albo"
 M00000000000012 "bar rd alum 1.000 dia"    round "aluminum"    1.000 0  0  inchs
 n/a "n/a" "" "" "b211" "qq-a-225/9" "7075" "" "" t nil nil nil nil nil "t73"
   "alco"
M00000000000013 "bar rd steel 1.125 dia"    round "steel"    1.125 0  0  inchs
 n/a "n/a" "" "" "a108" "" "" "" "" t nil nil nil nil nil "n/a"
"csuj"
M00000000000026 "flat stock steel .375 x 5.000"  flat "steel"    0  5.000 0.375 inchs
 n/a "n/a" "mil-s-11310" "1018" "a29" "qq-w-461" "" "" "" nil t nil nil nil nil "n/a"
   ""
M00000000000035 "flat stock alum .500 x .750"  flat "aluminum"    0  0.750 0.500 inchs
 n/a "n/a" "" "" "b211" "qq-a-200/8" "" "" "" t nil nil nil nil nil "t6511"
   ""
```

C. SAMPLE METCUT MATERIAL MAPPING CATALOG

```
catalog
{(:set-package "RAMP")}
{(:defcatalog *-METCUT-MATERIAL-MAPPING-*)}
{(:defline (:ramp-ID                ; A Unique Identifier
           :metcut-material-family  ; Material Family
           :metcut-material-id      ; Material ID
           :metcut-material-hc      ; Material Hardness & Condition
           ))}
{(:prepare-to-read)}

M00000000000001 "NICKEL ALLOYS, WROUGHT AND CAST -"          "MONEL ALLOY K500"
"150-320  HB - SOLUTION TREATED"
M00000000000002 "NICKEL ALLOYS, WROUGHT AND CAST -"          "MONEL ALLOY 400"
"115-240  HB - ANNEALED, COLD DRAWN OR CAST"
M00000000000003 "COPPER ALLOYS, CAST -"            "932"          "40-150   HB
500kg - AS CAST"
M00000000000004 "COPPER ALLOYS, CAST -"            "932"          "40-150   HB
500kg - AS CAST"
M00000000000005 "COPPER ALLOYS, CAST -"            "932"          "40-150   HB
500kg - AS CAST"
M00000000000006 "ALLOY STEELS, WROUGHT - Medium Carbon"      "4130"       "45-48
 HRC - QUENCHED AND TEMPERED"
M00000000000007 "ALLOY STEELS, WROUGHT - Medium Carbon"      "4140"       "275-325
 HB - NORMALIZED OR QUENCHED AND TEMPERED"
M00000000000008 "NICKEL ALLOYS, WROUGHT AND CAST -"          "MONEL ALLOY K500"
"150-320  HB - SOLUTION TREATED"
M00000000000009 "STAINLESS STEELS, WROUGHT - Martensitic"    "410"        "175-225
 HB - ANNEALED"
M00000000000010 "COPPER ALLOYS, CAST -"            "932"          "40-150   HB
500kg - AS CAST"
;M00000000000010 "POWDER METAL ALLOYS - Brasses"          "70CU-30ZN"    "35-81
 HRH - AS SINTERED"
M00000000000011 "ALUMINUM ALLOYS, WROUGHT -"              "6061"       "30-80
HB 500kg - COLD DRAWN"
M00000000000012 "ALUMINUM ALLOYS, WROUGHT -"              "7075"       "30-80
HB 500kg - COLD DRAWN"
M00000000000013 "CARBON STEELS, WROUGHT - Low Carbon"        "1018"       "85-125
 HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"
M00000000000014 "TITANIUM ALLOYS, WROUGHT - Commercially Pure"      "99.0"
"200-275  HB - ANNEALED"
M00000000000015 "CARBON STEELS, WROUGHT - Low Carbon"        "1018"       "85-125
 HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"
M00000000000016 "CARBON STEELS, WROUGHT - Low Carbon"        "1018"       "85-125
 HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"
M00000000000017 "STAINLESS STEELS, WROUGHT - Austenitic"     "302"        "225-275
 HB - COLD DRAWN"
 M00000000000018 "CARBON STEELS, WROUGHT - Low Carbon"       "1018"       "85-125
 HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"
M00000000000019 "FREE MACHINING STAINLESS STEELS, WROUGHT - Austenitic" "303"
"135-185  HB - ANNEALED"
M00000000000020 "CARBON STEELS, WROUGHT - Low Carbon"        "1018"       "85-125
 HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"
```

M00000000000021 "CARBON STEELS, WROUGHT - Low Carbon" HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"	"1018"	"85-125
M00000000000022 "CARBON STEELS, WROUGHT - Low Carbon" HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"	"1018"	"85-125
M00000000000023 "CARBON STEELS, WROUGHT - Low Carbon" HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"	"1018"	"85-125
M00000000000024 "CARBON STEELS, WROUGHT - Low Carbon" HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"	"1018"	"85-125
M00000000000025 "CARBON STEELS, WROUGHT - Low Carbon" HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"	"1018"	"85-125
M00000000000026 "CARBON STEELS, WROUGHT - Low Carbon" HB - HOT ROLLED, NORMALIZED, ANNEALED OR COLD DRAWN"	"1018"	"85-125
M00000000000027 "ALUMINUM ALLOYS, WROUGHT -" HB 500kg - COLD DRAWN"	"6061"	"30-80
M00000000000028 "ALUMINUM ALLOYS, WROUGHT -" HB 500kg - COLD DRAWN"	"6061"	"30-80

D. SAMPLE TIME STANDARDS CATALOG

```
catalog
{(:set-package "RAMP")}
{(:defcatalog "-TIME-STANDARDS-")}
{(:defline (:index
            :workstation-id
            :time-type
            :constant?
            :variable-occurances?
            :time
            :description
            ))}
{(:prepare-to-read)}

1  tlbs01 o T   NIL  0.5 "Wand bar code."
2  tlbs01 s T   NIL  0.5 "Pull job data."
3  tlbs01 o T   NIL  2.0 "Read job data."
4  tlbs01 s NIL T    3.0 "Clean/assemble necessary tools (use with TC K40 system)."
5  tlbs01 s NIL T    5.0 "Clean/assemble necessary tools (use CV50 adapter, collets, and extension)."
6  tlbs01 s T   T    2.0 "Measure tool for length and dia."
7  tlbs01 s NIL NIL  1.0 "Carry tools from tool crib to TC."
8  tlbs01 s NIL NIL  1.0 "Carry tools from tool crib to HMC."
9  tlbs01 s NIL NIL  2.0 "Insert tools in TC."
10 tlbs01 s NIL NIL  2.0 "Insert tool in HMC."
11 tlbs01 s T   NIL  0.5 "Enter tool data measurements into CNC950."
12 tlbs01 s NIL NIL  3.0 "Carry unneeded tools from TC to tool crib."
13 tlbs01 s NIL NIL  1.0 "Carry unneeded tools from HMC to tool crib."

1  mtps01 s T   NIL  1.0 "Pull down job data."
2  mtps01 s T   NIL  0.5 "Read job data."
3  mtps01 m NIL NIL  5.0 "Move piece by jib crane."
4  mtps01 s T   NIL  5.0 "Clamp, position, and remove remnant."
5  mtps01 o T   NIL  0.5 "Start and stop saw."
6  mtps01 s T   NIL  5.0 "Deburr saw cut & sweep chips -- leave job completed."
7  mtps01 s T   NIL  1.0 "Inspect piece with machinist square."
8  mtps01 s T   NIL  0.5 "Apply bar code."
9  mtps01 s T   NIL  0.5 "Wand in barcoded material."
10 mtps01 s NIL NIL  5.0 "Move by jib crane to staging conveyor for fixturing."

1  fixs02 s NIL NIL  0.5 "Wand bar code."
2  fixs02 s T   NIL  0.5 "Pull down job data."
3  fixs02 s T   NIL  3.5 "Read job data & graphics."
4  fixs02 s NIL NIL 15.0 "Change chucks with jib crane."
5  fixs02 o NIL NIL 20.0 "Bore the chuck jaws."
6  fixs02 s NIL NIL  0.5 "Blow off chuck."
7  fixs02 s NIL NIL 12.0 "Change 3-jaw chuck (assemble/disassemble)."
8  fixs02 s NIL NIL 15.0 "Change 4-jaw chuck (assemble/disassemble)."
9  fixs02 s NIL NIL  0.4 "Placing by hand to 3-jaw chuck."
10 fixs02 s NIL NIL  1.3 "Placing by hand to 4-jaw chuck."
11 fixs02 s NIL NIL  2.5 "Place by jib crane in 3-jaw chuck."
12 fixs02 s NIL NIL  3.5 "Place by jib crane in 4-jaw chuck."
13 fixs02 s T   NIL  0.5 "Clear chips away from part."
```

14 fixs02 s NIL NIL 0.4 "Remove part from chuck by hand."
15 fixs02 s NIL NIL 2.0 "Remove part from chuck with jib crane."
16 fixs02 s T NIL 0.5 "Apply bar code."

1 tncs01 o NIL NIL 0.5 "Wand bar code."
2 tncs01 o NIL NIL 2.0 "Read instructions and graphics."
3 tncs01 o NIL NIL 2.0 "Clear chips as necessary."
4 tncs01 o NIL NIL 0.5 "Delete old NC program from TC."
5 tncs01 o NIL NIL 2.0 "Download NC program to TC."
6 tncs01 o NIL NIL 1.0 "Home position machine and start cycle."

1 fixs01 s T NIL 0.5 "Wand bar code."
2 fixs01 s T NIL 0.5 "Pull down job data."
3 fixs01 s T NIL 3.0 "Read instruction and graphics."
4 fixs01 s NIL NIL 1.0 "Turn fixture stand."
5 fixs01 s NIL NIL 12.0 "Change angle stand with jib crane."
6 fixs01 s NIL NIL 8.0 "Position 3-jaw chuck on fixture."
7 fixs01 s NIL NIL 0.2 "Position dowel in fixture plate."
8 fixs01 s NIL NIL 1.0 "Position block/edge clamp in fixture."
9 fixs01 s NIL NIL 1.5 "Position hold-down clamps on fixture."
10 fixs01 s NIL NIL 2.0 "Position vise on fixture."
11 fixs01 s NIL NIL 0.5 "Place part in vise."
12 fixs01 s NIL NIL 0.4 "Place part into fixture by hand."
13 fixs01 s NIL NIL 0.5 "Place part in 3-jaw chuck."
14 fixs01 s NIL NIL 2.5 "Place part into fixture by jib crane."
15 fixs01 s NIL NIL 0.5 "Blow chips from holes."
16 fixs01 s NIL NIL 0.5 "Clear chips from fixture."

1 hmcs01 o T NIL 0.5 "Wand bar code."
2 hmcs01 o T NIL 0.5 "Pull down job data."
3 hmcs01 o T NIL 3.0 "Read instructions and graphics."
4 hmcs01 o NIL NIL 2.0 "Blow off."
5 hmcs01 s NIL NIL 2.0 "Blow off fixture."
6 hmcs01 s NIL NIL 1.0 "Turn fixture stand."
7 hmcs01 o NIL NIL 0.5 "Delete old NC program from HMC."
8 hmcs01 o NIL NIL 2.0 "Download NC program to HMC."
9 hmcs01 o NIL NIL 1.0 "Home-position machine and start cycle."
10 hmcs01 m NIL NIL 0.3 "Remove 35lb part from fixture by hand."
11 hmcs01 m NIL NIL 2.0 "Remove part with jib crane."
12 hmcs01 m NIL NIL 0.1 "Remove dowel -- clean and store."
13 hmcs01 m NIL NIL 1.1 "Remove block/edge clamp -- clean and store."
14 hmcs01 m NIL NIL 0.8 "Remove hold-down/edge clamp."
15 hmcs01 m NIL NIL 2.5 "Remove vise -- clean and store."
16 hmcs01 m NIL NIL 9.0 "Remove 3-jaw chuck -- clean and store."
17 hmcs01 m NIL NIL 0.4 "Remove part from vise (leave vise mounted) -- clean and store."
18 hmcs01 m NIL NIL 0.4 "Remove part from chuck (leave chuck mounted) -- clean and store."
19 hmcs01 s NIL NIL 0.5 "Apply bar code."

1 dbrs01 o T NIL 0.5 "Wand in bar code."
2 dbrs01 o T NIL 0.5 "Pull down job data."
3 dbrs01 o T NIL 2.0 "Read job data."
4 dbrs01 m NIL NIL 5.0 "Move 50lb part by hand from conveyor to bench and back."
5 dbrs01 m NIL NIL 10.0 "Move 50lb part by jib crane from conveyor to bench and back."
6 dbrs01 o T NIL 1.0 "Visual inspection and handling to feel burrs."

7 dbrs01 o NIL NIL 0.5 "Blow chips from holes."
8 dbrs01 o NIL T 0.15 "Deburr through-holes for 0.25 dia."
9 dbrs01 o NIL T 0.4 "Deburr through-holes for 1.0 dia."
10 dbrs01 o NIL T 1.0 "Deburr through-holes for 2.0 dia."
11 dbrs01 o NIL T 3.0 "Deburr through-holes for 4.0 dia."
12 dbrs01 o NIL T 0.3 "Deburr blind-holes for 0.25 dia."
13 dbrs01 o NIL T 0.4 "Deburr blind-holes for 1.0 dia."
14 dbrs01 o NIL T 0.6 "Deburr blind-holes for 2.0 dia."
15 dbrs01 o NIL T 2.0 "Deburr blind-holes for 4.0 dia."
16 dbrs01 o NIL T .15 "Deburr straight and curved edges (0.15 min/inch)."
17 dbrs01 o NIL NIL 1.0 "Mark part (inscribe)."
18 dbrs01 o T NIL 1.0 "Wipe and wash off part."
19 dbrs01 o NIL NIL 0.5 "Apply new bar code."
20 dbrs01 o T NIL 0.5 "Wand bar code when job is complete."

1 inss01 o T NIL 0.5 "Wand in bar code."
2 inss01 o T NIL 1.0 "Use visual check to inspect."
3 inss01 o NIL T 1.0 "Use plug gauge to inspect."
4 inss01 o NIL T 1.0 "Use snap gauge to inspect."
5 inss01 o NIL T 1.0 "Use gauge blocks to inspect."
6 inss01 o NIL T 1.0 "Use feeler gauge to inspect."
7 inss01 o NIL T 1.0 "Use height gauge to inspect."
8 inss01 o NIL T 1.0 "Use inside mike for (2-3\"dim)."
9 inss01 o NIL T 1.0 "Use inside mike (3-10\") to inspect."
10 inss01 o NIL T 1.0 "Use indicator with stand to inspect per dim."
11 inss01 o NIL T 1.0 "Use machinist square to inspect."
12 inss01 o NIL T 1.0 "Use machinist scale to inspect."
13 inss01 o NIL T 1.0 "Use outside mike to inspect."
14 inss01 o NIL T 1.0 "Use dial calipers inside/outside to inspect."
15 inss01 o NIL T 1.0 "Use depth mike to inspect."
16 inss01 o NIL T 1.0 "Use thread mike to inspect hole dim."
17 inss01 o NIL T 3.0 "Use profilometer to inspect per dim."
18 inss01 o NIL T 1.0 "Use protractor per dim to inspect."
19 inss01 o NIL T 1.0 "Use radius gauge per dim to inspect."
20 inss01 o NIL T 1.0 "Use taper gauge per dim to inspect."
21 inss01 o T NIL 1.0 "Input dimensions to workstation after inspection."

1 prps01 o T NIL 0.5 "Wand bar code."
2 prps01 o T NIL 0.5 "Pull job data."
3 prps01 s T NIL 2.0 "Read instructions."
4 prps01 o T NIL 5.0 "Coat part with protectant."
5 prps01 o T NIL 2.0 "Wrap part and place in container."
6 prps01 o T NIL 5.0 "Print-out part quality per part and place in shipping container."
7 prps01 o T NIL 2.0 "Completely pack and secure container."
8 prps01 o T NIL 0.5 "Label container for shipping."
9 prps01 m NIL NIL 5.0 "Hand lift 50lb carton and place."
10 prps01 m NIL NIL 10.0 "Move 50lb finished carton by jib crane."

1 exts01 o T NIL 0.0 "External Processing"

1 recs01 o T NIL 0.0 "Receive part following External Processing"

E. FIXTURING CONCEPTS LIST

```
(defparameter *FIXTURE-CONCEPTS-LIST*
        '((:chuck-inner-front
            "Interior Chucking on Front of Part"
            "Turning"
            ("Mount soft jaws on chuck."
             "Insert shop aid and clamp with jaws."
             "Machine jaws to the specified diameter and depth."
             "Unclamp jaws and remove shop aid."
             "Insert workpiece and clamp with jaws."))
          (:chuck-inner-rear
           "Interior Chucking on Rear of Part"
           "Turning"
           ("Mount soft jaws on chuck."
            "Insert shop aid and clamp with jaws."
            "Machine jaws to the specified diameter and depth."
            "Unclamp jaws and remove shop aid."
            "Insert workpiece and clamp with jaws."))
          (:chuck-outer-front
           "Exterior Chucking on Front of Part"
           "Turning"
           ("Mount soft jaws on chuck."
            "Insert shop aid and clamp with jaws."
            "Machine jaws to the specified diameter and depth."
            "Unclamp jaws and remove shop aid."
            "Insert workpiece and clamp with jaws."))
          (:chuck-outer-rear
           "Exterior Chucking on Rear of Part"
           "Turning"
           ("Mount soft jaws on chuck."
            "Insert shop aid and clamp with jaws."
            "Machine jaws to the specified diameter and depth."
            "Unclamp jaws and remove shop aid."
            "Insert workpiece and clamp with jaws."))
          (:chuck-inner-front
           "Interior Chucking on Front of Part"
           "Milling")
          (:chuck-inner-rear
           "Interior Chucking on Rear of Part"
           "Milling")
          (:chuck-outer-front
           "Exterior Chucking on Front of Part"
           "Milling")
          (:chuck-outer-rear
           "Exterior Chucking on Rear of Part"
           "Milling")
          (:center-hole-dowel-location-3-Pt
           "Center Hole Dowel Location - 3 Pt Contact"
           "Milling"
           ("Locate dowel in grid plate."
            "Locate 3 point contacts in grid plate."
            "Locate clamps in grid plate."
            "Mount workpiece onto fixture."
```

```
  "Tighten clamps and ensure rigidity of fixture."))
(:center-hole-dowel-location-4-Pt
  "Center Hole Dowel Location - 4 Pt Contact"
  "Milling"
  ("Locate dowel in grid plate."
   "Locate 4 point contacts in grid plate."
   "Locate clamps in grid plate."
   "Mount workpiece onto fixture."
   "Tighten clamps and ensure rigidity of fi ture."))
;;added by bill for testing 9/11/91
(:chuck-inner-front
  "Interior Chucking on Front of Part"
  "Holemaking")
(:chuck-inner-rear
  "Interior Chucking on Rear of Part"
  "Holemaking")
(:chuck-outer-front
  "Exterior Chucking on Front of Part"
  "Holemaking")
(:chuck-outer-rear
  "Exterior Chucking on Rear of Part"
  "Holemaking")
(:center-hole-dowel-location-3-Pt
  "Center Hole Dowel Location - 3 Pt Contact"
  "Holemaking"
  ("Locate dowel in grid plate."
   "Locate 3 point contacts in grid plate."
   "Locate clamps in grid plate."
   "Mount workpiece onto fixture."
   "Tighten clamps and ensure rigidity of fixture."))
(:center-hole-dowel-location-4-Pt
  "Center Hole Dowel Location - 4 Pt Contact"
  "Holemaking"
  ("Locate dowel in grid plate."
   "Locate 4 point contacts in grid plate."
   "Locate clamps in grid plate."
   "Mount workpiece onto fixture."
   "Tighten clamps and ensure rigidity of fixture."))
))
```

This page intentionally left blank.

APPENDIX V

A. SAMPLE PROCESS PLAN DESCRIPTIONS FILE

```
################################################################################
#
#                         Operation 1:  Tool Preparation
#
#     Wand bar code.                                                    0:00:30
#     Pull job data.                                                    0:00:30
#     Read job data.                                                    0:02:00
#     Measure tool for length and dia.                                  0:02:00
#     Enter tool data measurements into CNC950.                         0:00:30
#                                                                      ========
#                                           Total Operation Time ===>  0:05:30
#
################################################################################

################################################################################
#
#                       Operation 2:  Material Preparation
#
#
#     Cut Raw Stock to a length of 5.375 inches.                        0:15:00
#
#                                                                      ========
#                                           Total Operation Time ===>  0:15:00
#
################################################################################


################################################################################
#
#                          Operation 3:  TRC Fixturing
#
#     Read job data & graphics.                                         0:03:30
#     Placing by hand to 3-jaw chuck.                                   0:00:24
#
#                                                                      ========
#                                           Total Operation Time ===>  0:03:54
#
################################################################################
```

```
##########################################################################
#
#                        Operation 4:  Small Turning Center
#
#
# a.   Feature:      Circular Face at (0.0 5.125 0.0)                   0:00:53
#      Operation:    Semi-Fin Face  (1 of 2)
#      Tool:         TRN-754-03
#      Speed: 443 rpm     Number of passes:  2
#      Feed:         3.101 ipm     Time per pass:    27 seconds
#
# b.   Feature:      Circular Face at (0.0 5.125 0.0)                   0:00:28
#      Operation:    Finish Face  (2 of 2)
#      Tool:         TRN-754-03
#      Speed: 451 rpm     Number of passes:  1
#      Feed:         2.932 ipm     Time per pass:    28 seconds
#
# c.   Feature:      Outer Diameter With No Restricted Sides at
#                    (0.0 2.562 0.                                       0:01:42
#      Operation:    Normal Rough Turn  (1 of 2)
#      Tool:         TRN-754-03
#      Speed: 375 rpm     Number of passes:  2
#      Feed:         6.0 ipm       Time per pass:    51 seconds
#
# d.   Feature:      Outer Diameter With No Restricted Sides at
#                    (0.0 2.562 0                                        0:01:18
#      Operation:    Finish Turn  (2 of 2)
#      Tool:         TRN-754-01
#      Speed: 610 rpm     Number of passes:  1
#      Feed:         3.965 ipm     Time per pass:    1 minute, 18 seconds
#
# e.   Feature:      Circular Od Edge Flat at (0.0 5.109 0.0)           0:00:01
#      Operation:    Finish Contour  (1 of 1)
#      Tool:         TRN-758-03
#      Speed: 622 rpm     Number of passes:  2
#      Feed:         4.043 ipm     Time per pass:    0  second
#
# f.   Feature:      Square Circular Od Groove at (0.0 2.57 0.0)        0:00:11
#      Operation:    Finish Groove  (1 of 1)
#      Tool:         Grooving Tool
#      Speed: 325 rpm     Number of passes:  1
#      Feed:         0.975 ipm     Time per pass:    11 seconds
#
#                                                                       ========
#
#                                          Total Operation Time ===>    0:04:34
#
##########################################################################
```

```
##################################################################################
#
#                         Operation 5:  Inspection
#
#
#    Use visual check to inspect.                                         0:01:00
#    Input dimensions to workstation after inspection.                    0:01:00
#
#                                                                         =======
#
#                                        Total Operation Time ===>        0:02:00
#
##################################################################################


##################################################################################
#
#                     Operation 6:  Small Turning Center
#
#
# a.   Feature:     Circular Face at (0.0 0.0 0.0)                        0:00:53
#      Operation:   Semi-Fin Face  (1 of 2)
#      Tool:        TRN-754-03
#      Speed: 443 rpm      Number of passes:  2
#      Feed:        3.101 ipm      Time per pass:    27 seconds
#
#
# b.   Feature:     Circular Face at (0.0 0.0 0.0)                        0:00:28
#      Operation:   Finish Face  (2 of 2)
#      Tool:        TRN-754-03
#      Speed: 451 rpm      Number of passes:  1
#      Feed:        2.932 ipm      Time per pass:    28 seconds
#
#
# c.   Feature:     Circular Od Edge Flat at (0.0 0.016 0.0)              0:00:01
#      Operation:   Finish Contour  (1 of 1)
#      Tool:        TRN-758-03
#      Speed: 622 rpm      Number of passes:  2
#      Feed:        4.043 ipm      Time per pass:     0 second
#
#                                                                         =======
#
#                                        Total  Operation Time ===>       0:01:22
#
##################################################################################
```

Operation 7: Deburr and Wash

Wipe and wash off part.	0:01:00
Visual inspection and handling to feel burrs.	0:01:00
Deburr straight and curved edges (0.15 min/inch). (2 x 9 seconds)	0:00:18
Mark part (inscribe).	0:01:00
	========
Total Operation Time ===>	0:03:18

Operation 8: Inspection

Wand in bar code.	0:00:30
Use radius gauge per dim to inspect. (3 x 1 minute)	0:03:00
Input dimensions to workstation after inspection.	0:01:00
	========
Total Operation Time ===>	0:04:30

Operation 9: Packaging

Read instructions.	0:02:00
Coat part with protectant.	0:05:00
Wrap part and place in container.	0:02:00
	========
Total Operation Time ===>	0:09:00

B. SAMPLE PROCESS PLAN ROUTING FILE

0000000000010nonecvsmp zz	720.0	0.0	0.0
0001011000010nonetlbs01aa	0.0	30.0	0.0
0002021000010cartmtps01ab	16.0	14.0	5.0
0003051000010handtncs01ad	19.0	30.0	5.0
0004061000010cartdbrs01af	7.0	0.0	5.0
0005031000010cartfixs01ag	0.0	38.0	5.0
0006041000010handhmcs01ah	20.0	35.0	5.0
0007030900010cartfixs01ai	0.0	17.0	5.0
0008060900010handdbrs01aj	6.0	0.0	5.0
0009081000010cartinss01ak	20.0	0.0	5.0
0010101000010noneexts01al	0.0	0.0	0.0
0011111000010cartrecs01am	15.0	0.0	5.0
0012080900010handinss01an	16.0	0.0	5.0
0013100900010noneexts01ao	0.0	0.0	0.0
0014110900010cartrecs01ap	15.0	0.0	5.0
0015080800010handinss01aq	6.0	0.0	5.0
0016091000010noneprps01ar	16.0	2.0	0.0

C. SAMPLE TOOL DETAIL FILE

Turning Id, drawing #	: TRN-754-03 TRN-1
Holder Mfr/Desig	: KENNAMETAL KCLPR-164D
Operation Category	: Face & Turn or Contour
Lead Angle	: -5.0 Degrees
Insert Identifier	: INS-0197
Insert Mfr/Desig	: KENNAMETAL CPC-422
Insert Shape	: 80 Deg Diamond
Chip Control	: None
Tool Material Class	: Uncoated Carbide
Industry Grade	: C2
Manufacturers Grade	: K68
Nose Size/Style	: 0.0310 Radius
Insert IC (size)	: 0.500
Hand of Tool	: Right Hand
Rake Style	: Positive
Point Angle	: 80
Overall Length	: 6.00
Shank Type	: Square or Rectangular Shank
Shank Height or Dia	: 1.000
Shank Width	: 1.000
Num of Insert Pockets	: 1
Insert Seat Desig	: SM-369
Insert Pin Desig	: STC-4
Insert Clamp Desig	: CK-12
Insert Thickness	: 0.125
Insert Rake Style	: Neutral
Insert Relief Style	: Standard Positive - 9 to 17 Degrees
Insert Relief Angle	: 11
Cutting Edge Condition	: Sharp Cutting Edge - Not Honed
Surface Condition	: Ground - All Surfaces
# of Index. Cut Edges	: 4

Thread Type Category	: Not Applicable
Thread Appl Category	: Not Applicable
External Thread Pitch	: Min : 0.00 Max : 0.00
Internal Thread Pitch	: Min : 0.00 Max : 0.00
Coolant Fed	: No

APPENDIX VI

SAMPLE FEATURE INSTANCE ATTRIBUTE VALUES

The following is a list of PDES Interpreter attribute values for a SQUARE-CIRCULAR-OD-GROOVE. A similar set of attributes would be computed for each feature variant.

#<SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION 4580>, an object of flavor
PDES INTERPRETER::SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION,
has instance variable values:
SHADOW-OBJECT: :UNBOUND
INHERIT-MESSAGE: :PDES INTERPRETER-GROUP--PDES INTERPRETER-OBJECT
AGGREGATE: #<RULE::SERIES PDES INTERPRETER-OBJECT 140041573>
PARENT: #<PDES INTERPRETER::PDES INTERPRETER-GROUP
17083>
INDEX: 0
UNIQUE-ID: 17095
DISPLAY-CONTROLS: (:COLOR :CYAN)
CENTER: #<ARRAY 3 simple 140040306>
INVERSE-ORIENTATION: :UNBOUND
NORMAL-CENTER: :UNBOUND
OBLIQUENESS: NIL
OBLIQUENESS-CONTEXT: :UNBOUND
ORIENTATION: (:RIGHT :REAR :TOP)
HEIGHT: :UNBOUND
LENGTH: :UNBOUND
WIDTH: :UNBOUND
HELP-ITEMS: NIL
HELP-NAME: NIL
HELP-SPEC: ("(4580)
SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION ->
SQUARE-CIRCULAR-OD-GROOVE"
(NIL "There is no special help for this object."))
DESCRIPTION-ITEMS: ((:ID "ID") (:MFG-TYPE "Mfg. Feature Type")
:DIMENSIONAL-TOLERANCES :GEOMETRIC-TOLERANCES : NOTES
(:SURFACE-FINISH-ROUGHNESS "Surface Finish (microns)") (:CENTER-POINT "Center")
:UNTRANSFORMED-PDES-X :UNTRANSFORMED-PDES-Y :UNTRANSFORMED-PDES-Z :PDES-X
:PDES-Y :PDES-Z :ICAD-X
:ICAD-Y :ICAD-Z :ORIENTATION-MATRIX :UNTRANSFORMED-PDES-LOCATION-POINT
:PDES-LOCATION-POINT
:CENTER-POINT :TOLERANCED-BASE-WIDTH :TOLERANCED-ANGLE1 :TOLERANCED-ANGLE2
:TOLERANCED-RADIUS1
:TOLERANCED-RADIUS2 :TOLERANCED-PATH-SIZE :TOLERANCED-DEPTH
:SYMMETRIC-ABOUT-PART-AXIS?
:PROFILE-OPENING-RELATIONSHIP-TO-AXIS :ICAD-Y :ICAD-Z :ICAD-X :CENTER-POINT)

DESCRIPTION-NAME: PDES
INTERPRETER::SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION
BASE-DESCRIPTION-ITEMS:((:MFG-TYPE "Mfg. Feature Type") :DIMENSIONAL-TOLERANCES
:GEOMETRIC-TOLERANCES :NOTES
(:SURFACE-FINISH-ROUGHNESS "Surface Finish (microns)") (:CENTER-POINT "Center")
:UNTRANSFORMED-PDES-X :UNTRANSFORMED-PDES-Y :UNTRANSFORMED-PDES-Z :PDES-X
:PDES-Y :PDES-Z :ICAD-X
:ICAD-Y :ICAD-Z :ORIENTATION-MATRIX :UNTRANSFORMED-PDES-LOCATION-POINT
:PDES-LOCATION-POINT
:CENTER-POINT)
1.3.2.2.1.5.1.4. 4580
MFG-CATEGORY: :FEATURES
MFG-TYPE: SQUARE-CIRCULAR-OD-GROOVE
PDES-ENTITY: #<PDES Entity FORM-FEATURE 4580>
SPECIFIC-DESCRIPTION-ITEMS: (:TOLERANCED-BASE-WIDTH
:TOLERANCED-ANGLE1 :TOLERANCED-ANGLE2 :TOLERANCED-RADIUS1
:TOLERANCED-RADIUS2 :TOLERANCED-PATH-SIZE :TOLERANCED-DEPTH
:SYMMETRIC-ABOUT-PART-AXIS?
:PROFILE-OPENING-RELATIONSHIP-TO-AXIS :ICAD-Y :ICAD-Z :ICAD-X :CENTER-POINT)
STRINGS-FOR-DISPLAY: "(4580)
SQUARE-U-PROFILE-COMPLETE-CIRCULAR-PATH-DEPRESSION ->
SQUARE-CIRCULAR-OD-GROOVE"
CENTER-POINT: #<ARRAY 3 simple 140040230>
ICAD-X: #<ARRAY 3 simple 140041356>
ICAD-Y: #<ARRAY 3 simple 140040234>
ICAD-Z: #<ARRAY 3 simple 140041451>
ORIENTATION-MATRIX: #<ARRAY 3x3 simple 140042002>
PDES-AXIS2-PLACEMENT: #<PDES Entity AXIS2-PLACEMENT 4300>
PDES-LOCATION-POINT: #<ARRAY 3 simple 140040230>
PDES-X: #<ARRAY 3 simple 140041451>
PDES-Y: #<ARRAY 3 simple 401005013>
PDES-Z: #<ARRAY 3 simple 140040234>
SPATIAL-MIXIN-DESCRIPTION-ITEMS: (:UNTRANSFORMED-PDES-X
:UNTRANSFORMED-PDES-Y :UNTRANSFORMED-PDES-Z :PDES-X :PDES-Y :PDES-Z
:ICAD-X :ICAD-Y :ICAD-Z :ORIENTATION-MATRIX :UNTRANSFORMED-PDES-LOCATION-POINT
:PDES-LOCATION-POINT :CENTER-POINT)
UNTRANSFORMED-PDES-LOCATION-POINT: #<ARRAY 3 simple 140040077>
UNTRANSFORMED-PDES-X: #<ARRAY 3 simple 140042023>
UNTRANSFORMED-PDES-Y: #<ARRAY 3 simple 401004724>
UNTRANSFORMED-PDES-Z: #<ARRAY 3 simple 140040103>
DIMENSIONAL-TOLERANCES: (#<LOCATION-DIMENSION 4720>
#<LOCATION-DIMENSION 4680>)
GEOMETRIC-TOLERANCES: NIL
NOTE-IDS: (3700)
NOTES: (#<FEATURE-MACHINING-ALLOWED-SURFACE-FINISH 3700>)
SURFACE-FINISH: #<FEATURE-MACHINING-ALLOWED-SURFACE-FINISH 3700>
SURFACE-FINISH-ROUGHNESS: 63
TOLERANCE-IDS: (4720 4680)
PDES-FEATURE-VOLUME: #<PDES Entity ALONG-FEATURE-SWEEP 4540>
END-BOUND-FEATURE: #<CYLINDRICAL-FACE 3460>
PDES-END-BOUND: #<PDES Entity MAXIMAL-AREA-SHAPE-ELEMENT 3820>
PDES-SWEEP-PATH: #<PDES Entity COMPLETE-CIRCULAR-FEATURE-SWEEP-PATH
4520>

PATH-DESCRIPTION-ITEMS: (:TOLERANCED-PATH-SIZE :TOLERANCED-DEPTH
:SYMMETRIC-ABOUT-PART-AXIS?
:PROFILE-OPENING-RELATIONSHIP-TO-AXIS :ICAD-Y :ICAD-Z :ICAD-X :CENTER-POINT)
PROFILE-OPENING-RELATIONSHIP-TO-AXIS:: AWAY
SYMMETRIC-ABOUT-PART-AXIS?: T
TOLERANCED-DEPTH: 0.18500000000000005 +/- 0.0f0
TOLERANCED-PATH-SIZE: 1.625 +/- 0.01 (FULL)
PDES-SWEEP-PROFILE: #<PDES Entity SQUARE-U-FEATURE-SWEEP-PROFILE 4440>
PROFILE-DESCRIPTION-ITEMS: (:TOLERANCED-BASE-WIDTH :TOLERANCED-ANGLE1
:TOLERANCED-ANGLE2 :TOLERANCED-RADIUS1 :TOLERANCED-RADIUS2)
TOLERANCED-ANGLE1: 90.0 +/- 0.5 (FULL)
TOLERANCED-ANGLE2: 90.0 +/- 0.5 (FULL)
TOLERANCED-BASE-WIDTH: 0.359375 +0.007812 -0.0
TOLERANCED-RADIUS1: 0.093 +/- 0.01 (HALF)
TOLERANCED-RADIUS2: 0.093 +/- 0.01 (HALF)

This page intentionally left blank.

APPENDIX VII

SAMPLE IGES FILE

```
Generated with ICAD CAD Output Tools 8.7 by LHazzard                S 1
,,15HUnknown Product,6HSTREAM,20HICAD IGES Output 9.1,14HIGES Version 3,G 1
32,38,7,308,15,15HUnknown Product,1.0,1,2HIN,32767,32.767,          G 2
13H920122.162927,1.0E-4,10000.0,8HLHazzard,4HICAD,4,0;              G 3
 124    1    1    1    0    0    0   000000000D   1
 124    0    0    1    0             CIRCLE    0D   2
 100    2    1    1    0    0    1   000000000D   3
 100    0    0    1    0             CIRCLE    0D   4
 124    3    1    1    0    0    0   000000000D   5
 124    0    0    1    0             CIRCLE    0D   6
 100    4    1    1    0    0    5   000000000D   7
 100    0    0    1    0             CIRCLE    0D   8
 100    5    1    1    0    0    5   000000000D   9
 100    0    0    1    0             CIRCLE    0D  10
 110    6    1    1    0    0    0   000000000D  11
 110    0    0    1    0               LINE    0D  12
 110    7    1    1    0    0    0   000000000D  13
 110    0    0    1    0               LINE    0D  14
 110    8    1    1    0    0    0   000000000D  15
 110    0    0    1    0               LINE    0D  16
 110    9    1    1    0    0    0   000000000D  17
 110    0    0    1    0               LINE    0D  18
 100   10    1    1    0    0    1   000000000D  19
 100    0    0    1    0             CIRCLE    0D  20
 110   11    1    1    0    0    0   000000000D  21
 110    0    0    2    0               LINE    0D  22
 110   13    1    1    0    0    0   000000000D  23
 110    0    0    2    0               LINE    0D  24
 110   15    1    1    0    0    0   000000000D  25
 110    0    0    2    0               LINE    0D  26
 110   17    1    1    0    0    0   000000000D  27
 110    0    0    1    0               LINE    0D  28
 100   18    1    1    0    0    1   000000000D  29
 100    0    0    1    0             CIRCLE    0D  30
 110   19    1    1    0    0    0   000000000D  31
 110    0    0    2    0               LINE    0D  32
 110   21    1    1    0    0    0   000000000D  33
 110    0    0    1    0               LINE    0D  34
 110   22    1    1    0    0    0   000000000D  35
 110    0    0    1    0               LINE    0D  36
 110   23    1    1    0    0    0   000000000D  37
 110    0    0    1    0               LINE    0D  38
 100   24    1    1    0    0    5   000000000D  39
```

100	0	0	1	0			CIRCLE	0D	40
100	25	1	1	0	0	5	000000000D		41
100	0	0	1	0			CIRCLE	0D	42
124	26	1	1	0	0	0	000000000D		43
124	0	0	2	0			ARC	0D	44
100	28	1	1	0	0	43	000000000D		45
100	0	0	2	0			ARC	0D	46
124	30	1	1	0	0	0	000000000D		47
124	0	0	2	0			ARC	0D	48
100	32	1	1	0	0	47	000000000D		49
100	0	0	2	0			ARC	0D	50
124	34	1	1	0	0	0	000000000D		51
124	0	0	2	0			ARC	0D	52
100	36	1	1	0	0	51	000000000D		53
100	0	0	2	0			ARC	0D	54
124	38	1	1	0	0	0	000000000D		55
124	0	0	2	0			ARC	0D	56
100	40	1	1	0	0	55	000000000D		57
100	0	0	2	0			ARC	0D	58
100	42	1	1	0	0	5	000000000D		59
100	0	0	1	0			CIRCLE	0D	60
110	43	1	1	0	0	0	000000000D		61
110	0	0	2	0			LINE	0D	62
110	45	1	1	0	0	0	000000000D		63
110	0	0	1	0			LINE	0D	64
110	46	1	1	0	0	0	000000000D		65
110	0	0	2	0			LINE	0D	66
110	48	1	1	0	0	0	000000000D		67
110	0	0	1	0			LINE	0D	68
100	49	1	1	0	0	5	000000000D		69
100	0	0	1	0			CIRCLE	0D	70
124	50	1	1	0	0	0	000000000D		71
124	0	0	2	0			ARC	0D	72
100	52	1	1	0	0	71	000000000D		73
100	0	0	2	0			ARC	0D	74
124	54	1	1	0	0	0	000000000D		75
124	0	0	2	0			ARC	0D	76
100	56	1	1	0	0	75	000000000D		77
100	0	0	2	0			ARC	0D	78
124	58	1	1	0	0	0	000000000D		79
124	0	0	2	0			ARC	0D	80
100	60	1	1	0	0	79	000000000D		81
100	0	0	2	0			ARC	0D	82
124	62	1	1	0	0	0	000000000D		83
124	0	0	1	0			ARC	0D	84
100	63	1	1	0	0	83	000000000D		85
100	0	0	2	0			ARC	0D	86
100	65	1	1	0	0	1	000000000D		87
100	0	0	1	0			CIRCLE	0D	88
110	66	1	1	0	0	0	000000000D		89
110	0	0	2	0			LINE	0D	90
110	68	1	1	0	0	0	000000000D		91
110	0	0	2	0			LINE	0D	92
110	70	1	1	0	0	0	000000000D		93

```
 110     0     0     1     0              LINE      0D    94
 110    71     1     1     0     0     0     000000000D    95
 110     0     0     1     0              LINE      0D    96
 402    72     1     1     0     0     0     000000000D    97
 402     0     0     2     7          MFG-PART        0D    98
124,1.0,0.0,0.0,0.0,0.0,0.0,1.0,0.0,0.0,-1.0,0.0,0.0;               1P  1
100,5.125,0.0,0.0,0.96625,0.0,0.96625,0.0;                          3P  2
124,-1.0,0.0,-0.0,0.0,0.0,0.0,-1.0,0.0,0.0,-1.0,-0.0,0.0;           5P  3
 100,0.0,0.0,0.0,0.96625,0.0,0.96625,0.0;                           7P  4
100,-5.09375,0.0,0.0,0.9975,0.0,0.9975,0.0;                         9P  5
110,0.96625,5.125,-1.7749139E-16,0.9975,5.09375,-1.8323173E-16;      11P  6
110,-1.1832759E-16,5.125,-0.96625,-3.312306E-24,5.09375,-0.9975;     13P  7
110,-0.96625,5.125,5.9163796E-17,-0.9975,5.09375,6.107724E-17;       15P  8
110,0.0,5.125,0.96625,0.0,5.09375,0.9975;                          17P  9
100,0.03125,0.0,0.0,0.9975,0.0,0.9975,0.0;                         19P 10
110,-0.96625,5.3482097E-10,-1.7749139E-16,-0.9975,0.03125,          21P 11
-1.8323173E-16;                                                   21P 12
110,1.8476369E-24,5.3482097E-10,-0.96625,1.2215449E-16,0.03125,     23P 13
-0.9975;                                                         23P 14
110,0.96625,5.3482097E-10,5.9163796E-17,0.9975,0.03125,             25P 15
6.107724E-17;                                                     25P 16
110,0.0,5.3482097E-10,0.96625,0.0,0.03125,0.9975;                    27P 17
100,2.7499995,0.0,0.0,0.9975,0.0,0.9975,0.0;                       29P 18
110,-0.9975,2.7499995,-1.8323173E-16,-0.9975,5.09375,              31P 19
-1.8323173E-16;                                                   31P 20
110,0.0,2.7499995,-0.9975,0.0,5.09375,-0.9975;                      33P 21
110,0.9975,2.7499995,6.107724E-17,0.9975,5.09375,6.107724E-17;      35P 22
110,0.0,2.7499995,0.9975,0.0,5.09375,0.9975;                        37P 23
100,-2.7499995,0.0,0.0,0.9055,0.0,0.9055,0.0;                      39P 24
100,-2.6569996,0.0,0.0,0.8125,0.0,0.8125,0.0;                      41P 25
124,0.99999994,-0.0,1.8369095E-16,0.0,0.0,0.99999994,0.0,0.0,      43P 26
-1.8369102E-16,0.0,1.0,0.0;                                       43P 27
100,1.323489E-23,-0.90549994,2.6569993,-0.81249994,2.6569993,       45P 28
-0.90549994,2.7499995;                                            45P 29
124,-1.2246066E-16,0.0,1.0,0.0,0.0,0.99999994,0.0,0.0,             47P 30
 -0.99999994,0.0,-1.2246064E-16,0.0;                               47P 31
100,0.0,-0.90549994,2.6569993,-0.81249994,2.6569993,-0.90549994,    49P 32
2.7499995;                                                       49P 33
124,-0.99999994,0.0,-6.123032E-17,0.0,0.0,0.99999994,-0.0,0.0,      51P 34
6.123033E-17,-0.0,-1.0,0.0;                                       51P 35
100,0.0,-0.90549994,2.6569993,-0.81249994,2.6569993,-0.90549994,    53P 36
2.7499995;                                                       53P 37
124,0.0,0.0,-1.0,0.0,0.0,0.99999994,-0.0,0.0,0.99999994,0.0,       55P 38
-0.0,0.0;                                                        55P 39
100,0.0,-0.90549994,2.6569993,-0.81249994,2.6569993,-0.90549994,    57P 40
2.7499995;                                                       57P 41
100,-2.4836245,0.0,0.0,0.8125,0.0,0.8125,0.0;                      59P 42
110,-0.8125,2.4836245,-1.492489E-16,-0.8125,2.6569996,              61P 43
-1.492489E-16;                                                    61P 44
110,0.0,2.4836245,-0.8125,0.0,2.6569996,-0.8125;                    63P 45
110,0.8125,2.4836245,4.9749633E-17,0.8125,2.6569996,                65P 46
4.9749633E-17;                                                    65P 47
110,0.0,2.4836245,0.8125,0.0,2.6569996,0.8125;                      67P 48
100,-2.3906245,0.0,0.0,0.9055,0.0,0.9055,0.0;                      69P 49
```